

TUGAS AKHIR - TE 141599

DESAIN SISTEM TOP-UP SALDO PADA *INTELLIGENT TRANSPORT SYSTEM* KOTA SURABAYA

Ulfi Romadhani Rosadiana
NRP 0711144000044

Dosen Pembimbing
Dr. Ir. Achmad Affandi, DEA
Ir. Gatot Kusrahardjo, MT

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



TUGAS AKHIR - TE 141599

**DESAIN SISTEM TOP-UP SALDO PADA *INTELLIGENT*
TRANSPORT SYSTEM KOTA SURABAYA**

Ulfi Romadhani Rosadiana
NRP 0711144000044

Dosen Pembimbing
Dr. Ir. Achmad Affandi, DEA
Ir. Gatot Kusrahardjo, MT

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



FINAL PROJECT - TE 141599

DESIGN OF PASSENGER BALANCE TOP-UP SYSTEM INTELLIGENT TRANSPORT SYSTEM FOR SURABAYA CITY

Ulfi Romadhani Rosadiana
NRP 0711144000044

Supervisors
Dr. Ir. Achmad Affandi, DEA
Ir. Gatot Kusrahardjo, MT

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi keseluruhan Tugas akhir saya dengan judul “**DESAIN SISTEM TOP-UP SALDO PADA *INTELLIGENT TRANSPORT SYSTEM* KOTA SURABAYA**” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juni 2018

Ulfi Romadhani Rosadiana
0711144000044

**DESAIN SISTEM TOP-UP SALDO PADA
INTELLIGENT TRANSPORT SYSTEM KOTA
SURABAYA**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada**

**Bidang Studi Telekomunikasi Multimedia
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Ir. Achmad Affandi, DEA.
NIP. 1965101419900021001

Ir. Gatot Kusrahardjo, MT.
NIP. 195904281986011001



DESAIN SISTEM TOP-UP SALDO PENUMPANG PADA *INTELLIGENT TRANSPORT SYSTEM* KOTA SURABAYA

Nama mahasiswa : Ulfi Romadhani Rosadiana
Dosen Pembimbing I : Dr. Ir. Achmad Affandi, DEA
Dosen Pembimbing II : Ir. Gatot Kusraharjo, MT

Abstrak

Surabaya memiliki program untuk menuju kota cerdas dengan masuk ke dimensi baru yaitu Angkutan Masal Cepat (AMC) yang modern dan berkesinambungan. AMC Surabaya akan mendukung dimensi teknologi informasi dan komunikasi untuk dapat memungkinkan melakukan pemindahan informasi secara *real-time* mengenai posisi armada dan manajemen pendapatan yang berpusat pada sistem kartu.

AMC Surabaya berkilatkan Intelligent Transport System yang dalam hal ini masuk dalam kategori Advanced Public Transportation Sytem(APTS) dimungkinkan untuk dapat melakukan Electronic Fare Payment. Dalam pembayaran elektronik ini salah satu hal yang perlu dikembangkan adalah mengenai top-up sistem pada kartu penumpang. Desain mengenai sistem kerja dari top-up saldo akan dimulai dari metode apa saja yang dapat dipakai, sistem pembayaran, aplikasi yang dapat menunjang sistem top-up saldo kartu, hingga perangkat server yang digunakan.

Pada tahap pengujian dibagi menjadi empat pengujian. Pada pengujian sistem aplikasi menggunakan Borland Delphi 6 dan website AMC telah dapat berfungsi sesuai rancangan. Pengujian jaringan mengambil data throughput dan delay dengan throughput terbesar bernilai 6096,76 bits/s dan delay terendah sebesar 116,96 ms. Semua nilai delay yang didapat juga diatas 150 ms sesuai dengan standar ITU-T *recommendation* Y1541 dimana nilai *delay* <150 ms berada pada kondisi yang *excellent*. Terakhir pengujian server didapatkan dengan user antar 500-4000 server masih dapat bekerja dengan performa yang baik.

Kata kunci: *Intelligent Transport System (ITS)*, manajemen pendapatan, *Top-up Saldo*

PASANGER BALANCE TOP-UP SYSTEM DESIGN AT INTELEGENT TRANSPORT SYSTEM SURABAYA CITY

Student Name : Ulfi Romadhani Rosadiana
Supervisor I : Dr. Ir. Achmad Affandi, DEA
Supervisor II : Ir. Gatot Kusrahardjo, MT

Abstract

Surabaya has a program to get to the smart city by entering into a new dimension of Modern Mass Transportation (AMC) which is modern and sustainable. AMC Surabaya will support the dimension of information and communication technology to enable real-time information transfer of fleet position and revenue management centered on the card system.

AMC Surabaya affiliated Intelligent Transport System which in this case included in the category of Advanced Public Transportation Sytem (APTS) is possible to do Electronic Fare Payment. In this electronic payment one of the things that need to be developed is about top-up system on the passenger card. The design of the top-up balance system will be started from what methods can be used, payment systems, applications that can support the card's top-up balance system, to the server device used.

The testing phase is divided into four tests. In testing the application system using Borland Delphi 6 and AMC website has been able to function according to the design. Network testing takes data throughput and delay with the largest throughput of 6096.76 bits / s and the lowest delay of 116.96 ms. All delay values are also above 150 ms according to ITU-T recommendation standard Y1541 where the delay value <150 ms is in excellent condition. Last test server got with user between 500-4000 server still can work with good performance.

Keywords: Intelligent Transport System (ITS), revenue management, Top-up Balance

Kata Pengantar

Puji Syukur penulis panjatkan kehadiran Allah SWT atas segala Rahmat, Karunia, dan Petunjuk yang telah dilimpahkan-Nya sehingga penulis mampu menyelesaikan tugas akhir dengan judul “DESAIN SISTEM TOP-UP SALDO PENUNJANG PADA *INTELLIGENT TRANSPORT SYSTEM* KOTA SURABAYA”.

Tugas Akhir ini disusun sebagai salah satu persyaratan untuk menyelesaikan jenjang pendidikan S1 pada Bidang Studi Telekomunikasi Multimedia, Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember. Atas selesainya penyusunan tugas akhir ini, penulis mengucapkan terima kasih kepada:

1. Tuhan Yang Maha Kuasa atas limpahan rahmat dan karunia-Nya.
2. Ibu dan Ayah penulis atas doa dan cinta yang tak henti pada penulis dalam keadaan apapun.
3. Bapak Dr. Ir. Achmad Affandi, DEA dan Ir. Gatot Kusrahajra, MT selaku dosen pembimbing yang telah memberikan arahan dan bimbingan selama proses penyelesaian tugas akhir ini.
4. Seluruh dosen dan karyawan Departemen Teknik Elektro ITS yang telah memberikan banyak ilmu dan menciptakan suasana belajar yang luar biasa.
5. Teman-teman seperjuangan e54 yang telah menemani dan memberikan dukungan selama masa kuliah sampai penyusunan tugas akhir ini.

Penulis telah berusaha maksimal dalam penyusunan tugas akhir ini. Namun tetap besar harapan penulis untuk menerima saran dan kritik untuk perbaikan dan pengembangan tugas akhir ini. Semoga tugas akhir ini dapat memberikan manfaat yang luas.

Surabaya, 3 Juni 2018

Penulis

Daftar Isi

Abstrak.....	i
Abstract.....	iii
Kata Pengantar	v
Daftar Isi	vii
Daftar Gambar	xi
Daftar Tabel	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	2
1.5 Metodologi	2
1.6 Sistematika Penulisan	4
1.7 Manfaat.....	5
BAB II TINJAUAN PUSTAKA.....	7
2.1 ITS (Intelligent Tranport System).....	7
2.1.1 Kategori ITS.....	8
2.2 Angkutan Masa Cepat AMC	9
2.2.1 Sistem Jaringan Komunikasi	9
2.3 OBU.....	10
2.4 Client-server	11
2.5 Web Service.....	12

2.6	Jaringan Seluler	13
2.7	Borland Delphi 6	13
2.8	XAMPP	15
2.8.1	Apache	15
2.8.2	MySQL	15
2.8.3	PHP	16
2.8.4	PHPMyAdmin	16
2.9	Quality of Service (QoS).....	17
2.9.1	Throughput	17
2.9.2	Delay.....	18
2.10	Wireshark	19
BAB III PERANCANGAN DAN IMPLEMENTASI SISTEM.....		21
3.1	Perancangan Sistem Management Revenue ITS Surabaya ..	21
3.2	Perancangan Desain Sistem Top-Up Saldo	23
3.2.1	Penetapan Metode Sistem Top-Up Saldo	25
3.2.2	Penetapan Tarif Top-Up Saldo Kartu	26
3.2.3	Metode Pembayaran Top-up Saldo.....	26
3.2.4	Struktur Sistem Jaringan Komunikasi Top-up Saldo	26
3.2.5	Alur Perencanaan Sistem Komunikasi.....	27
3.3	Perangkat <i>Server</i> yang Digunakan	28
3.4	Pembuatan Aplikasi Sistem Top-Up pada OBU	29
3.4.1	Rancangan Pembuatan Aplikasi Sistem Top-Up	29
3.4.2	Langkah-Langkah Pembuatan Aplikasi Sistem Top-Up..	30
3.5	Pembuatan Aplikasi Sistem Top-Up Berbasis Web	35
3.5.1	Rancangan Aplikasi Sistem Top-Up	36
3.5.2	Langkah – Langkah Pembuatan Aplikasi Sistem Top-up	37

3.6	Perancangan <i>Database</i> PHPMyAdmin dan <i>Report System</i> .	44
3.7	Skenario Pengujian Sistem	45
3.7.1	Parameter Pengujian Realisasi Sistem	45
3.7.2	Pengujian Aplikasi Top-up Saldo Menggunakan OBU ..	46
3.7.3	Pengujian Aplikasi Top-up Saldo Menggunakan Website	48
3.7.4	Pengujian Implementasi Jaringan OBU dan Website.....	49
3.7.5	Pengujian Kinerja Server	52
BAB IV HASIL PENGUJIAN DAN ANALISIS		55
4.1	Hasil Pembuatan Aplikasi Sistem Top-Up Saldo	55
4.1.1	Hasil Tampilan Aplikasi Top-Up pada OBU	55
4.1.2	Hasil Pembuatan Aplikasi Top-Up Berbasis WEB	57
4.2	Hasil Pengujian Sistem Topup Berbasis OBU	59
4.2.1	Hasil Pengujian Operasional Sistem Top-up.....	59
4.2.2	Hasil pengujian Performansi Jaringan Sistem Topup.....	60
4.2	Hasil Pengujian Sistem Topup Berbasis Website.....	62
4.2.1	Hasil Pengujian Operasional Sistem Top-Up.....	62
4.2.2	Hasil Pengujian Performansi Sistem Top-Up.....	63
4.3	Pengujian Kinerja Server AMC Surabaya	64
BAB V PENUTUP.....		65
5.1	Kesimpulan.....	65
5.2	Saran.....	65
DAFTAR PUSTAKA		67
LAMPIRAN.....		69
RIWAYAT HIDUP.....		91

Halaman ini sengaja dikosongkan

Daftar Gambar

Gambar 1.1 Metodologi.....	4
Gambar 2.1 Komponen Intelligent Transport System	7
Gambar 2.2 Desain Manajemen Armada AMC.....	9
Gambar 2.3 Rancangan arsitektur sistem jaringan OBU	11
Gambar 2.4 Arsitektur client-server	12
Gambar 2.5 Borland Delphi 6.....	14
Gambar 2.6 XAMPP	15
Gambar 2.7 PHPMyAdmin	17
Gambar 2.8 Wireshark.....	19
Gambar 3.1 Sistem Menejemen Pendapatan AMC Surabaya.....	21
Gambar 3.2 Alur Sistem Top-Up Saldo	23
Gambar 3.3 Metode Sistem Top-Up Saldo	25
Gambar 3.4 Diagram Blok Sistem Jaringan Komunikasi Top-Up Saldo	26
Gambar 3.5 Diagram Alur Komunikasi Jaringan Terintegrasi Sistem Topup	27
Gambar 3.6 Informasi Server	28
Gambar 3.7 Diagram Alur Pembuatan Sistem Top-Up.....	29
Gambar 3.8 Rancangan Aplikasi Sistem Top-Up.....	30
Gambar 3.9 Tampilan Aplikasi Top-Up Sistem pada OBU	35
Gambar 3.10 Diagram Alur Pembuatan Sistem Top-Up.....	37
Gambar 3.11 Tampilan Awal Website	44
Gambar 3.12 Database Top-Up Sistem	44
Gambar 3.13 Laman webreport ITS	45
Gambar 3.14 Flowchart Pengujian Aplikasi menggunakan OBU	47
Gambar 3.15 Implementasi Sistem Jaringan	49
Gambar 3.16 Flowchart Pengujian Jaringan.....	50
Gambar 3.17 tampilan awal wireshark	51
Gambar 3.18 Tampilan pengukuran wireshark.....	51
Gambar 3.19 Tampilan untuk melihat nilai throughput.....	52
Gambar 3.20 Cntoh Pengujian Jaringan Berbasis Web	52
Gambar 4.1 Tampilan Awal Aplikasi Top-Up	55
Gambar 4.2 Tampilan Aplikasi Hasil Respon Server Setelah Tapping dan Pilihan Nominal Top-up.....	56
Gambar 4.3 Tampilan Aplikasi Setelah Pengisian Saldo	56

Gambar 4.4 Tampilan Halaman Awal Website.....57

Gambar 4.5 Tampilan Halaman Cek No. ID dan Kategori Pemilik
Kartu57

Gambar 4.6 Tampilan Laman Top-Up Bila No. ID dan Kategori Benar
.....58

Gambar 4.7 Tampilan Laman Saat Isian No. ID dan Kategori Salah...58

Gambar 4.8 Tampilan Top-Up Berhasil.....58

Gambar 4.9 Hasil Throughput pada OBU.....61

Gambar 4.10 Hasil Delay pada OBU61

Gambar 4.11 Grafik Throughput.....63

Gambar 4.12 Grafik Delay63

Daftar Tabel

Tabel 2.1 Standarisasi Delay ITU-T.....	18
Tabel 3.1 Target Uji	46
Tabel 4.1 Hasil Operasional Sistem Topup	59
Tabel 4.2 Hasil Pengujian Jaringan	60
Tabel 4.3 Hasil Operasional Sistem Top Up Website	62
Tabel 4.4 Hasil Pengujian Server	64

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kota Surabaya saat ini sedang mengembangkan sistem Angkutan Masal Cerdas (AMC) berbasis *Intelligent Transport System* (ITS) untuk bias menjadi kota cerdas (*smartcity*). AMC Surabaya akan mendukung adanya dimensi teknologi informasi dan komunikasi untuk memungkinkan melakukan pemindahan informasi secara *real-time*. AMC Surabaya akan masuk dalam kategori *Advanced Public Transportation System* (APTS) yang mana akan memungkinkan melakukan pengiriman posisi armada secara berkala dan melakukan pembayaran transportasi secara elektronik (*e-ticket*).

Mekanisme pembayaran transportasi secara elektronik akan berkesinambungan dengan sistem manajemen pendapatan AMC Surabaya. Sistem manajemen pendapatan sendiri masih memerlukan banyak pengembangan. Manajemen pendapatan akan difokuskan pada pihak penumpang untuk sistem *E-Ticketing*. Penumpang akan diharuskan untuk memiliki kartu RFID khusus yang dikeluarkan oleh pihak AMC Surabaya untuk memenuhi segala kebutuhan transaksi mulai dari pembayaran ketika menaiki armada angkutan masal cepat Kota Surabaya ataupun top-up saldo yaitu pengisian kembali saldo pada kartu AMC Kota Surabaya. Dalam menjalankan kedua hal tersebut akan dapat dimonitoring secara *real-time* oleh pihak *server* TMC yang akan memudahkan dalam memantau sistem manajemen pendapatan sendiri.

Pada Tugas Akhir ini, sistem yang didesain akan memberikan gambaran kegiatan yang akan dilakukan untuk menyelesaikan masalah perencanaan sistem top-up saldo penumpang AMC Kota Surabaya. Rencana sistem top-up saldo sendiri akan dibuat mulai dari rancangan umum metode top-up, pembayaran, maupun ongkos top-up. Kemudian akan dilanjutkan pada mesin top-up untuk desain aplikasi sistem top-up mulai dari yang berbasis OBU maupun berbasis web. Tak hanya itu, rancangan desain top-up juga akan mencakup pada server yang dipilih yang nantinya berhubungan dengan database untuk rekam jejak transaksi.

1.2 Perumusan Masalah

Berdasarkan uraian yang tergambar pada latar belakang, maka rumuskan permasalahan dari penelitian ini yaitu :

1. Perencanaan sistem top-up saldo di Kota Surabaya, khususnya dalam hal pengisian ulang atau top-up saldo penumpang angkutan masal cepat Kota Surabaya.
2. Melakukan pemindahan informasi serta mendapatkan nilai *Throughput* dan *Delay* dari sistem yang telah dirancang

1.3 Batasan Masalah

Batasan masalah pada penelitian ini yaitu :

1. Sistem top-up melalui bank service diabaikan.
2. Sistem transportasi yang digunakan adalah bus kota Surabaya.

1.4 Tujuan

Tujuan utama dari penelitian ini yaitu :

1. Menentukan pola sistem top-up saldo untuk diterapkan pada jaringan angkutan umum Kota Surabaya.
2. Membuat platform dan protokol untuk sistem top-up saldo penumpang

1.5 Metodologi

1. Study literatur

Pada tahap ini yang dilakukan adalah mempelajari buku, paper, serta referensi mengenai *Intelligent Transport System* (ITS), sistem top-up untuk manajemen pendapatan, sistem *server*, web service dan *database*. Tahap ini juga melakukan pembelajaran lebih lanjut mengenai penelitian sebelumnya mengenai manajemen pendapatan pada ITS. Tak hanya itu, pengkajian dari negara-negara maju yang telah lebih dulu menerapkan ITS juga akan dijadikan referensi

2. Perancangan sistem

Sistem yang dirancang adalah mengenai aturan system top-up secara umum kemudian dilanjutkan dengan pembuatan aplikasi yang sesuai. Komunikasi antara client yang akan melakukan top-up menuju server juga akan direncanakan dengan

cermat. Selain itu perencanaan jaringan juga dilakukan untuk memilih metode mana yang sesuai dengan Kota Surabaya.

3. Pengkodean

Tahap penerjemahan data dan pemecahan masalah yang dirancang dalam bahasa pemrograman. Program-program ini dimaksudkan untuk memberikan perintah pengisian ulang saldo dengan melakukan pembaruan data yang tersinkronisasi baik pada *database* ataupun pada data center OBU yang berada pada AMC Kota Surabaya.

4. Pengujian data

Melakukan pengujian fungsi dari sistem yang telah dibuat dan juga performa sistem tersebut. pengujian sangat penting untuk melihat apakah sistem dapat bekerja dengan baik dan beberapa hal yang mungkin perlu dibenahi.

5. Analisa data

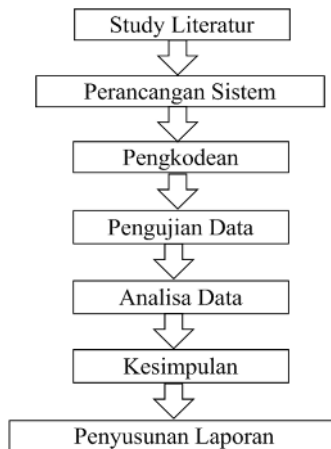
Menganalisa hasil dari pengujian data sehingga dapat ditentukan tingkat error, delay, dan kelayakan sistem yang telah dibuat. Tahap ini sangat perlu dilakukan untuk mengatasi kekurangan apa saja yang perlu ditinjau ulang pada sistem.

6. Kesimpulan

Memberikan kesimpulan dari pengujian system operasional aplikasi top-up dan kinerja jaringan maupun server. Kesimpulan juga berpatok pada analisa hasil data pengujian yang disesuaikan dengan standar ITU-T agar dapat dinilai kelayakan system yang dirancang dan direalisasikan untuk mengatasi masalah.

7. Penyusunan laporan

Menuangkan hasil kerja mulai dari latar belakang diangkatnya tema tugas akhir, rancangan sistem, pembuatan sistem hingga menjadi produk dan pengujian sistem ke dalam tulisan yang runtut. Tidak lupa melampirkan hasil sistem ketika bekerja dan juga analisa dari data sistem yang dibuat. kemudian menganalisisnya hingga menjadi laporan yang lengkap.



Gambar 0.1 Metodologi

1.6 Sistematika Penulisan

Sistematika penulisan Buku Tugas Akhir ini dibagi menjadi beberapa bab dengan rincian sebagai berikut :

Bab I Pendahuluan

Meguraikan latar belakang, perumusan masalah, batasan masalah, tujuan, metodologi, sistematika penulisan, dan manfaat yang berkaitan dengan pengerjaan Tugas Akhir.

Bab II Tinjauan Pustaka

Pada bab ini akan dibahas landasan teori mengenai konsep *Intelligent Transport System*, angkutan masal cepat Surabaya, *prototype On-Board Unit*, konsep *Client-Server*, *Web Service*, jaringan seluler, aplikasi pendukung sistem manajemen pendapatan dengan Borland Delphi 6 dan aplikasi-aplikasi pendukung untuk mendukung perangkat *server database* seperti XAMPP, Apache, MySQL, PHP, PHPMyAdmin, dan juga *Quality of Service* (QoS) dan aplikasi pendukung mengukur QoS.

Bab III Perancangan dan Implementasi Sistem

Pada bab perancangan dan implementasi sistem akan dijelaskan konsep yang akan diangkat pada Tugas Akhir ini mengenai perancangan sistem top-up saldo dan bagaimana pengimplemen-tasian sistem top-up

saldo penumpang yang mungkin cocok untuk Angkutan Masal Cepat Surabaya. Perancangan akan dimulai dengan perancangan desain sistem top-up secara umum kemudian penyediaan untuk perangkat server. Selanjutnya pembuatan aplikasi top-up berbasis OBU maupun berbasis Web untuk mesin/*interface* system, dilanjutkan dengan perancangan database. Terakhir membuat skenario pengujian sistem top-up untuk menilai kinerja sistem top-up maupun jaringan untuk pengantaran informasi dari client menuju server dan tes kinerja server sendiri.

Bab IV Hasil Pengujian dan Analisis

Bab empat akan menjabarkan hasil pembuatan aplikasi system top-up dan pengujiannya baik berbasis OBU ataupun website. Tidak hanya pengujian operasional namun pengujian jaringan juga akan dilakukan pada bab ini. Pengujian sistem dilakukan untuk menguji apakah sistem sesuai dengan rancangan atau tidak dan pengujian jaringan untuk menguji QoS dari jaringan yang telah diterapkan. Hasil pengujian server dengan beberapa scenario pengujian juga akan dipaparkan dilanjutkan analisis pengujian akan dikaitkan dengan standar yang telah ditetapkan ITU-T.

Bab V Penutup

Pada bab ini akan menampilkan kesimpulan dari seluruh rancangan, pembuatan, pengujian, hingga analisis yang didapat dari seluruh rangkaian pekerjaan. Selain itu, saran yang membangun untuk merealisasi dan mengembangkan sistem yang telah dirancang.

1.7 Manfaat

Manfaat dari tugas akhir ini adalah :

1. Peningkatan sistem kerja manajemen pendapatan pada ITS di Kota Surabaya.
2. Sebagai bahan kajian dan panduan untuk menentukan sistem top-up saldo penumpang angkutan masal cepat Kota Surabaya.
3. Meningkatkan kinerja operasional, layanan, dan persepsi terhadap angkutan umum kepada konsumen.
4. Memberikan kontribusi nyata terhadap sistem transportasi di Surabaya yang menuju kota percontohan bagi kota lain di Indonesia

Halaman ini sengaja dikosongkan

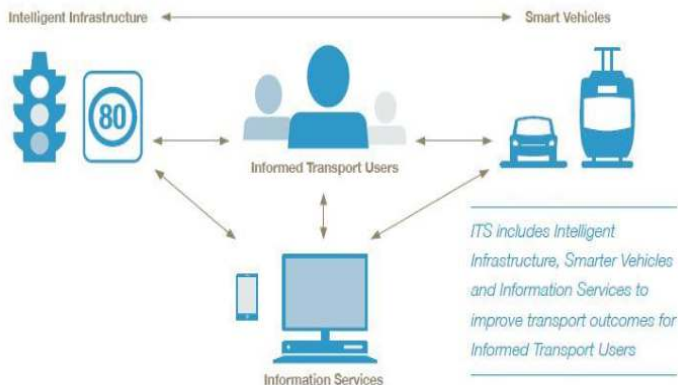
BAB II TINJAUAN PUSTAKA

2.1 ITS (Intelligent Transport System)

Intelligent Transport System atau ITS merupakan sistem transportasi cerdas yang berintegrasi antarsistem teknologi informasi dan komunikasi untuk mendukung dan mengoptimalkan semua model transportasi agar bekerja secara efektif dan efisien.

Dasar komponen ITS dapat dibagi menjadi tiga bidang yaitu :

- 1) Infrastruktur cerdas (Intelligent Infrastructure) seperti lampu-lampu lalu lintas, rambu-rambu jalan, dan tanda-tanda pesan untuk meningkatkan pengguna jalan, dll.
- 2) Kendaraan pintar (Smart Vehicle) seperti notifikasi kecelakaan otomatis, pembantu kecepatan cerdas, sistem navigasi GPS, peringatan tabrakan mundur dan depan, parking otomatis, dll.
- 3) Layanan Informasi (Information Services) informasi jadwal bus dan transportasi lain, sistem navigasi mobil, informasi kondisi lalu lintas untuk panduan sekitar daerah kemacetan, sistem pembayaran e-tiket, dll.



Gambar 0.1 Komponen Intelligent Transport System

2.1.1 Kategori ITS

Kategori ITS meliputi lima kelompok sistem pendukung diantaranya :

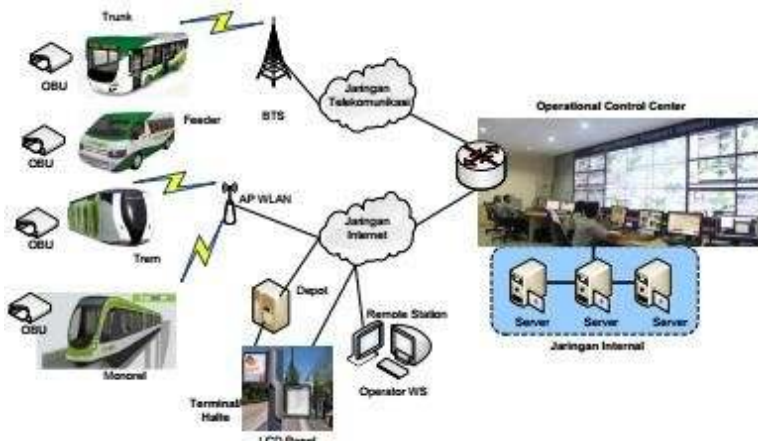
- *Advanced Traveler Information System (ATIS)*, sistem informasi canggih bagi pengendara, menyediakan pengendara informasi real-time, seperti rute transit dan jadwal; arah navigasi; dan informasi tentang penundaan akibat kemacetan, kecelakaan, kondisi cuaca, atau perbaikan jalan kerja
- *Advanced Transportation Management Systems (ATMS)*, ATMS, sistem manajemen transportasi canggih, meliputi perangkat kontrol lalu lintas, seperti lampu lalu lintas, meter jalan, tanda/pesan status lalu lintas yang dinamis, dan pusat-pusat operasi lalu lintas. *Traffic Operations Centers (TOCs)*, pusat operasi lalu lintas, merupakan pusat pengelolaan lalu lintas yang dioperasikan oleh kota mengandalkan teknologi informasi untuk menghubungkan sensor dan perangkat di pinggir jalan, probe kendaraan, kamera, tanda-tanda pesan, dan perangkat lain bersamasama untuk menciptakan pandangan yang terintegrasi dari aliran trafik dan mendeteksi kecelakaan, peristiwa cuaca yang berbahaya, atau bahaya jalan lainnya.
- *ITS-Enabled Transportation Pricing Systems, (ITS-ETPS)* ETPS, merupakan peran sentral dalam pembiayaan sistem transportasi. Contoh penerapan ETPS termasuk sistem pembayaran otomatis.
- *Advanced Public Transportation Systems (APTS)* APTS, sistem transportasi publik canggih, memungkinkan moda transpor publik seperti kereta, trem dan bus dapat melaporkan posisi mereka (*automatic vehicle location, AVL*), sehingga penumpang mendapat informasi status real-time mereka (informasi kedatangan dan keberangkatan) dan bagi pengelola dapat memonitor keberadaan aset mereka.
- *Fully integrated ITS*, termasuk integrasi *vehicle-to-infrastructure (V2I)* and *vehicle-to-vehicle (V2V)*, terkoneksi antar elemen dan aset sistem transportasi, seperti semua jenis sensor yang digunakan, lampu lalu lintas, dan kendaraan atau antar moda yang lain.

2.2 Angkutan Masa Cepat AMC

Dalam upaya mengatasi permasalahan kota Surabaya, diantaranya yaitu kemacetan, keterbatasan lahan untuk pembangunan jalan, perjalanan penumpang yang tidak efisien, dan meningkatnya polusi udara atas penggunaan kendaraan pribadi maka di butuhkan upaya penyediaan angkutan umum massal yang merupakan sarana transportasi berkelanjutan. Di sisi lain ketersediaan Angkutan Umum Massal Cepat (AMC) yang mampu menghubungkan prasarana transportasi utama di Kota Surabaya diharapkan mampu meningkatkan mobilitas orang secara efisien terhadap wilayah Surabaya dan sekitarnya sehingga dapat menarik investasi yang lebih besar [10]

2.2.1 Sistem Jaringan Komunikasi

Berangkat dari desain manajemen armada kota Surabaya dapat dibagi dan didefinisikan dari masing-masing bagian yang menjadi pekerjaan yang disusun sebagai tahap-tahap penerapan manajemen armada kota Surabaya. Berikut desain manajemen armada kota Surabaya:



Gambar 0.2 Desain Manajemen Armada AMC

Adapun tahap-tahap penerapan manajemen armada kota Surabaya sebagai berikut :

1. *Phase 1 CCROOM-Feeder Monitoring*

Pembangunan CCROOM yang disiapkan untuk mampu memonitoring *feeder vehicle* dan halte. Pembangunan ini membutuhkan pengadaan jaringan *hardware* dan *software* untuk internal CCROOM untuk meningkatkan kemampuan yang sekarang sudah ada.

2. *Phase 2 OBU Implementation*

Pada *phase* ini diawali dengan mengadakan *OBU* yang mempunyai modul GSM/GPS, *central unit*, piranti *human interface*, *diagnostic*

adapter, modul I/O, *unit power supply* dan baterai yang kemudian diimplementasi pada masing-masing *unit* armada dan dilakukan uji coba monitoring.

3. *Phase 3 Fleet Optimation*

Melakukan optimasi armada dengan menggunakan *software* yang mampu memberikan rekomendasi *routing* yang efisien dan berskala bisnis sehingga bisa memberikan penghematan bahan bakar, mengurangi waktu tempuh, dan waktu kerja sehingga dapat meningkatkan *revenue* dan mengurangi biaya operasi

4. *Phase 4 Operation and Maintenance*

Phase ini untuk memberi jaminan bahwa sistem informasi dapat berfungsi sesuai dengan yang direncanakan dan memberikan unjuk kerja yang optimal sampai sistem habis masa penggunaannya.

2.3 OBU

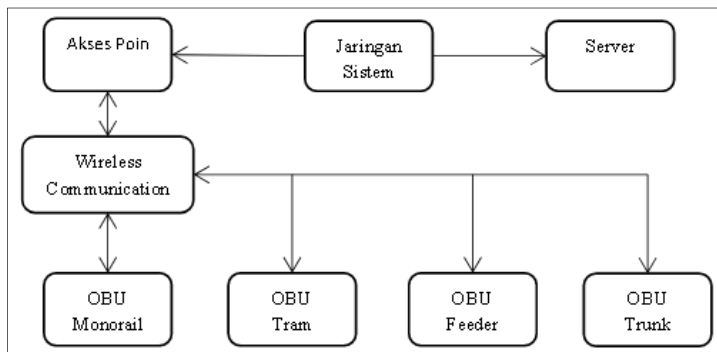
OBU adalah sistem ITS yang diletakkan di armada, berfungsi sebagai sebuah sistem pengendali masukan dan keluaran yang terkait fungsional manajemen armada, pendapatan (tiket), lalu lintas dan sistem darurat. Sehingga OBU akan dirancang mendukung beberapa fungsi berikut:

- Inisiasi dan penutupan sesi perjalanan armada AMC oleh pengemudi
- Identifikasi pengemudi dan penumpang, catuan ke sistem e-ticketing.

- Pemantauan posisi armada AMC
- Inisiasi sistem emergency armada
- Mendukung sistem VMS armada (display informasi perjalanan bagi penumpang armada AMC)
- Mendukung integrasi dengan sistem manajemen lalu lintas (Traffic Management Center) dan perjalanan armada AMC termasuk ke sistem VMS, sistempendapatan (e-ticketing).

Pada AMC Surabaya, akses poin Wi-Fi akan terdapat pada lokasi koridor pemberhentian. Akses poin tersebut direncanakan untuk membantu pengiriman data melalui jaringan internet. Hal tersebut berarti OBU yang digunakan harus memiliki kemampuan untuk mengakses Wi-Fi. Model pengiriman data ke *Server* akan lebih mudah dan dapat dilakukan seketika serta lebih hemat energi.

Gambar 2.3 menunjukkan diagram blok rancangan arsitektur sistem jaringan OBU dalam kerangka sistem ITS di Surabaya.



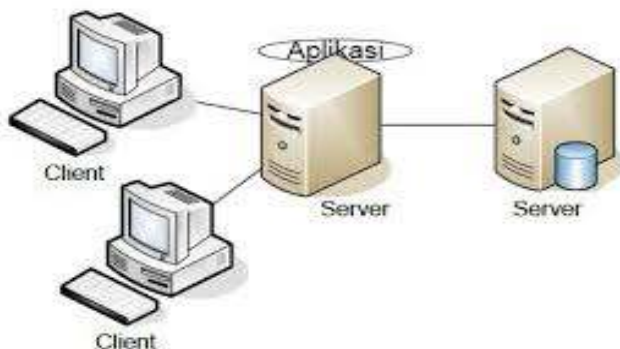
Gambar 0.3 Rancangan arsitektur sistem jaringan OBU

2.4 Client-server

Client-Server merupakan teknologi pendistribusian kerja aplikasi antara beberapa komputer yang dihubungkan oleh jaringan komunikasi yang mana salah satu akan bertindak untuk memberi layanan atau sebagai *server* dan yang lain akan menjadi peminta layanan atau disebut *client*.

Server adalah komputer yang memberikan pelayanan ke *client*. *Server* bersifat pasif, menunggu *request* dan apabila mendapat *request*, *server* akan memberikan balasan berupa *service*. Sedangkan

client adalah komputer yang mengakses beberapa *service* yang ada di *server*. Berkebalikan dengan *server*, *client* bersifat aktif dan bertindak sebagai pengirim *request* pada *server*.



Gambar 0.4 Arsitektur client-server

Struktur *client-server* memungkinkan untuk saling berbagi informasi dan sumber daya antar sistem misalnya file, prosesor, *disk space*, dan *peripheral* dapat berkolaborasi dan menyampaikan pesan antara prosesor dimana beberapa prosesor dijalankan sebagai *client* dan ada yang sebagai *server*. Dibutuhkan semacam mekanisme yang membuat setiap prosesor dapat mengetahui proses berjalannya jaringan mesin lain ketika dua prosesor sedang berkomunikasi melalui jaringan.

Interface pemrograman sederhana yang memiliki hubungan erat dengan proses komunikasi single dan multi mesin disediakan oleh *socket*. Tujuannya adalah untuk menyediakan sebuah metode umum IPC pada layer transport untuk mengetahui prosesor bekerja pada mesin yang sama.

2.5 Web Service

Web service akan menjadi aplikasi dengan sekumpulan data (*database*), perangkat lunak (*software*) atau bagian dari perangkat lunak yang dapat diakses secara remote oleh berbagai piranti yang disini dilakukan oleh pihak erver maupun petugas mesin top-up dengan sebuah perantara tertentu.

Secara umum, web service dapat diidentifikasi dengan menggunakan URL seperti hanya web pada umumnya. Berbeda dengan URL web pada umumnya, URL web service hanya mengandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi.

Web service dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah *database* ditanamkan, dibuat dalam bahasa apa, sebuah aplikasi yang mengkonsumsi data, dan di platform apa sebuah data itu dikonsumsi. Web service mampu menunjang interoperabilitas. Sehingga web service mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada.

2.6 Jaringan Seluler

Jaringan seluler menjadi suatu teknologi *wireless* yang digunakan untuk dapat terkoneksi ada jaringan internet sehingga memungkinkan pengiriman dan penerimaan ‘paket data. Untuk transmisi data ini menggunakan sistem CDMA, GSM, EDGE, 3G, HSPDA, 4G dan untuk kedepannya akan sangat memungkinkan jaringan 5G.

Jaringan *wireless* beroperasi dalam sebuah jaringan yang membagi kota atau wilayah kedalam sel-sel yang lebih kecil. Satu sel mencakup beberapa blok kota atau sampai 250 mil persegi. Setiap sel menggunakan sekumpulan frekuensi radio atau saluran-saluran untuk memberikan layanan di area spesifik. Kekuatan radio ini harus dikontrol untuk membatasi jangkauan geografis. Oleh Karena itu, frekuensi yang sama dapat digunakan kembali di sel terdekat. Maka banyak orang dapat melakukan percakapan secara simultan dalam sel yang berbeda di seluruh kota atau wilayah, meskipun mereka berada dalam satu saluran. Dalam setiap sel, terdapat stasiun dasar yang berisi antenna *wireless* dan perlengkapan radio lain. Antenna *wireless* dalam setiap sel akan menghubungkan sumber ke jaringan telepon lokal, internet, ataupun jaringan *wireless* lain.

2.7 Borland Delphi 6

Delphi umumnya digunakan untuk pengembangan aplikasi desktop dan enterprise berbasis *database*, namun sebagai perangkat pengembang yang bersifat *general-purpose* Delphi juga mampu dalam berbagai jenis proyek pengembangan *software*.

Delphi mendukung pengembangan aplikasi berbasis Rapid application development yang membawa fitur berupa kerangka/framework aplikasi serta aplikasi pendisain visual yang memungkinkan proses pengembangan sebuah program menjadi lebih cepat.

Delphi mendukung kompilasi antar platform, menghasilkan kode biner untuk Mac OS, Windows 32 bit, ataupun Windows 64 bit. Dengan menggunakan dialek object pascal, delphi memperkenalkan konsep VCL (Visual Component Library) sebagai basis kerangka kerja untuk modularisasi entitas-entitas baik yang bersifat visual ataupun non-visual dalam metodologi pemrograman berorientasi objek.

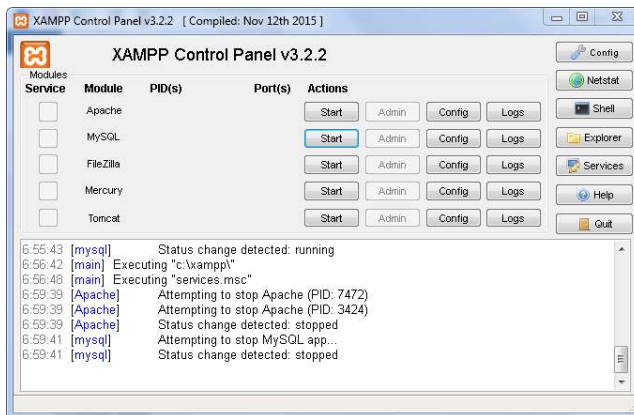
Dalam Borland Delphi 6 yang digunakan pada perancangan sistem ini dilengkapi ZeosLib. Zeos ini merupakan komponen tambahan yang dapat digunakan untuk penghubung dengan *database*, seperti MySQL, PostgreSQL, SQLite, maupun Oracle. ZeosLib sendiri dapat lebih memudahkan user dalam mengembangkan aplikasi desktop berbasis *database* sehingga tidak perlu menggunakan banyak item dan konfigurasi yang lebih rumit.



Gambar 0.5 Borland Delphi 6

2.8 XAMPP

XAMPP adalah perangkat lunak yang mendukung banyak sistem operasi dan kompilasi dari beberapa program. Fungsi XAMPP pada komputer adalah menjadikannya *server* yang berdiri sendiri (*localhost*). XAMPP sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. Dengan menggunakan XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server* Apache, PHP, dan MySQL secara manual karena XAMPP akan menginstalasi dan mengkonfigurasi secara otomatis. Dalam satu paket XAMPP terdapat lima *modul* yaitu : Apache, MySQL, File Zilla, Mercury dan Tomcat.



Gambar 0.6 XAMPP

2.8.1 Apache

Apache berfungsi untuk menampilkan halaman web yang sesuai dengan PHP yang telah dibuat. Apache bersifat *open source*, artinya siapa saja boleh menggunakan, mengambil, dan mengubah kode programnya.

2.8.2 MySQL

MySQL merupakan salah satu sistem manajemen *database* yang digunakan untuk membangun sebuah aplikasi berbasis *client-serve*. SQL (*Structures Query Language*) merupakan sebuah bahasa pemrograman yang digunakan untuk berkomunikasi dengan *server*.

MySQL bersifat *open source* dan *relational* yang artinya data-data yang dikelola akan diletakkan pada beberapa tabel terpisah sehingga proses manipulasi data akan menjadi lebih cepat. MySQL sendiri dapat digunakan untuk membuat dan mengelola *database* beserta isinya, seperti menambah, mengubah, menghapus data.

SQL sendiri terbagi menjadi beberapa bagian yaitu :

- DML (*Data Manipulation Language*) yaitu bahasa untuk memanipulasi data.
- DDL (*Data Definition Language*) yaitu bahasa untuk mendefinisikan struktur *database*, seperti CREATE, ALTER, dan DROP.
- *Transact SQL*, yaitu bahasa pemrograman yang diletakkan pada *database server*. Konten dari bahasa ini adalah menjaga *automacy* sebuah transaksi pertukaran data.

2.8.3 PHP

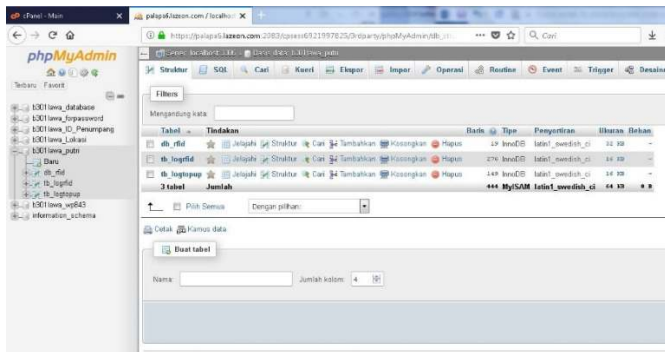
PHP merupakan bahasa pemrograman yang biasa digunakan dalam pembuatan aplikasi berbasis web yang bersifat *server-side* yang difungsikan untuk membangun suatu website dinamis. PHP menyatu dengan kode HTML, maksudnya adalah beda kondisi, HTML digunakan sebagai pembangun atau pondasi dari kerangka layout web, sedangkan PHP difungsikan sebagai prosesnya sehingga dengan adanya PHP tersebut, web akan sangat mudah di-*maintenance*. PHP berjalan pada sisi *server* sehingga PHP disebut juga sebagai bahasa *Server Side Scripting*. Artinya bahwa dalam setiap/untuk menjalankan PHP, wajib adanya *web server*.

PHP ini bersifat *open source* sehingga dapat dipakai secara cuma-cuma dan mampu lintas platform, yaitu dapat berjalan pada sistem operasi Windows maupun Linux. PHP juga dibangun sebagai modul pada *web server* apache dan sebagai binary yang dapat berjalan sebagai CGI. PHP dapat mendukung manajemen sistem *database* seperti MySQL, Oracle, Microsoft Access, Interbase, d-Base, dan PostgreSQL.

2.8.4 PHPMyAdmin

PHPMyAdmin sendiri merupakan perangkat lunak yang berfungsi untuk mengatur. Dengan PHPMyAdmin *user* dapat

dengan mudah membuat tabel, mengisi data, dan masih banyak yang lainnya yang tanpa harus hafal rumus pemrogramannya, namun cukup dengan mengisi tabel-tabel yang telah tersedia.



Gambar 0.7 PHPMYAdmin

2.9 Quality of Service (QoS)

Quality of Service (QoS) digunakan sebagai parameter untuk menentukan seberapa baik jaringan yang diberikan suatu layanan penyedia jaringan. QoS juga menjadi suatu usaha untuk mendefinisikan karakteristik dan sifat suatu layanan tersebut. Untuk mengukur kualitas layanan, beberapa aspek terkait dari layanan jaringan yang dipertimbangkan adalah packet loss, bit rate, *throughput*, *delay* transmisi, ketersediaan, jitter, dll.

Kualitas layanan diukur dengan standar dari *Telecommunication and Internet Protocol Harmonization Over Networks* (TIPHON) dan standar dari *International Telecommunication Union Telecommunication* (ITU- T) *recommendation* Y1541. Dalam tugas akhir ini parameter QoS yang akan dianalisis adalah *throughput* dan *delay* .

2.9.1 Throughput

Throughput diartikan sebagai laju data aktual per satuan waktu (*bits per second*). Biasanya *throughput* selalu dikaitkan dengan *bandwidth*. Karena *throughput* memang bisa disebut sebagai *bandwidth* dalam kondisi yang sebenarnya. *Bandwidth* lebih bersifat tetap sementara *throughput* sifatnya dinamis tergantung *trafik* yang sedang terjadi. Cara untuk menghitung *throughput* sebagai berikut :

$$a = \frac{n}{t} \quad (2.1)$$

Dimana : a = Throughput (bits/s)
 n = jumlah data yang diterima (bits)
 t = waktu pengiriman (s)

Dengan mendapat nilai throughput dari sebuah sistem yang sedang berjalan, kita bisa menentukan apakah sistem tersebut sudah memiliki kualitas yang baik atau belum. Apabila belum mendapatkan kualitas yang baik mungkin ada permasalahan dari sistem yang dibuat, atau pun bisa terjadi adanya permasalahan dalam jaringan yang dipakai.

2.9.2 Delay

Salah satu parameter untuk menilai QOS dari sebuah jaringan adalah *delay*. *Delay* atau waktu paket di dalam sistem adalah waktu sejak paket tiba kedalam sistem, sampai paket selesai ditransmisikan. Salah satu jenis *delay* adalah *delay* transmisi, yaitu waktu yang dibutuhkan untuk sebuah pengirim mengirimkan sebuah paket. *Delay* dapat dipengaruhi oleh kongesti, media fisik, jarak atau juga waktu proses yang lama. Berikut adalah parameter kualitas sebuah jaringan dilihat dari besarnya *delay* menurut ITU-T. Untuk menghitung *delay* dapat digunakan rumus berikut :

$$d = \frac{\Delta t}{n} \quad (2.2)$$

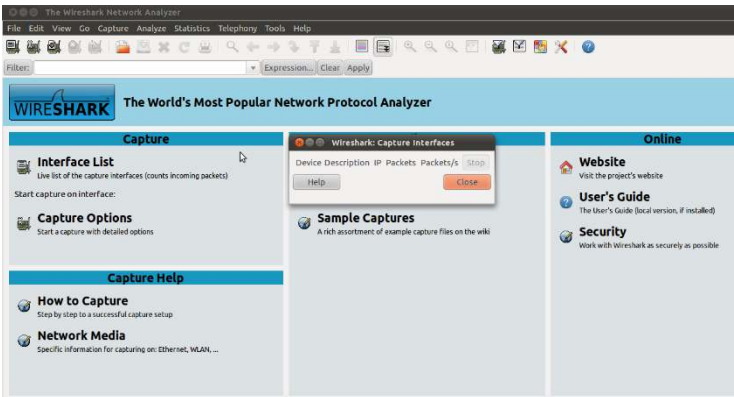
Dimana : d = delay (ms)
 Δt = waktu pengiriman antara paket awal dan akhir (ms)
 n = jumlah paket

Tabel 0.1 Standarisasi Delay ITU-T

No.	Kategori Delay	Besar Delay
1.	Excellent	<150 ms
2.	Good	150-300 ms
3.	Poor	300-450 ms
4.	Unnacceptable	>450 ms

2.10 Wireshark

Wireshark Network Protocol Analyzer adalah sebuah aplikasi perangkat lunak (software) yang digunakan untuk dapat melihat dan mencoba menangkap paket-paket jaringan dan berusaha untuk menampilkan semua informasi di paket tersebut sedetail mungkin. Open Source dari Wireshark menggunakan Graphical User Interface (GUI).



Gambar 0.8 Wireshark

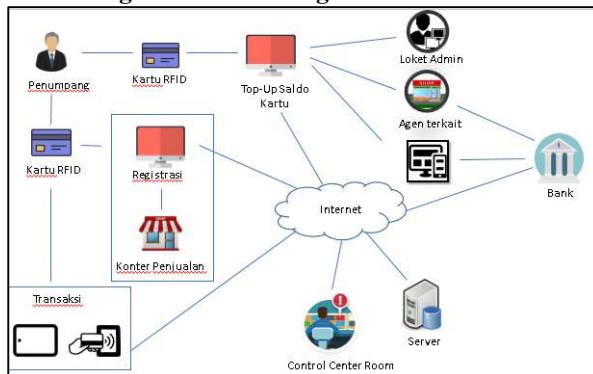
Halaman ini sengaja dikosongkan

BAB III

PERANCANGAN DAN IMPLEMENTASI SISTEM

Bab ini menjelaskan mengenai proses perancangan desain sistem top-up yang akan diimplementasikan di AMC Surabaya. Komunikasi pengiriman data dari *client* menuju *server* akan dilakukan dengan menggunakan dua metode yang berbeda yaitu Borland Delphi 6 dan web online dengan bahasa PHP. Alur yang menjelaskan garis besar proses yang dilakukan dalam melaksanakan tugas akhir sangat diperlukan untuk memudahkan penjelasan proses perancangan dan implementasi sistem ini.

3.1 Perancangan Sistem Management Revenue ITS Surabaya



Gambar 0.1 Sistem Menejemen Pendapatan AMC Surabaya

Kota Surabaya sedang mengembangkan sistem transportasi masal cerdas atau ITS yang terdiri atas bus, trem, dan monorail. Sistem transportasi cerdas pada Kota Surabaya dinamakan Angkutan Masal Cepat (AMC) Kota Surabaya ini memberikan layanan pembayaran menggunakan sistem *e-ticket* yang mana setiap penumpang akan menggunakan kartu RFID untuk setiap transaksi pada AMC Surabaya. Kartu RFID akan berisikan ID kartu penumpang untuk dapat melakukan pembayaran transportasi angkutan juga untuk top-up saldo kartu sendiri.

Kartu RFID menjadi bagian penting dalam perancangan sistem manajemen pendapatan karena semua pembayaran pada transportasi cerdas ini akan menggunakan kartu tersebut. Dari hal tersebut maka perlu perancangan matang pada bagian manajemen pendapatan karena seluruh transaksi akan dilakukan diluar perjalanan. Bagian sistem manajemen pendapatan dibagi menjadi tiga, yaitu:

1. Registrasi Kartu Baru
2. Pembayaran Perjalanan
3. Top-up Saldo Kartu Penumpang

Registrasi kartu baru adalah kegiatan penumpang dalam mendaftarkan diri untuk dapat memiliki kartu khusus yang digunakan untuk menaiki AMC Surabaya. Registrasi kartu baru diharuskan untuk mengisi beberapa data yang diperlukan agar pihak AMC Surabaya dapat mengenali dan memvalidasi pemegang kartu yang akan digunakan penumpang. Pada pendaftaran kartu baru dapat dilakukan melalui loket yang dijaga oleh petugas AMC Surabaya yang akan disediakan pada sebagian besar halte Kota Surabaya dan dapat pula dilakukan menggunakan mesin yang disediakan pada beberapa halte juga dimana pendaftaran ini dilakukan secara mandiri oleh penumpang namun tidak dapat melakukan pilihan kategori karena pada mesin hanya disediakan untuk kategori Dewasa.

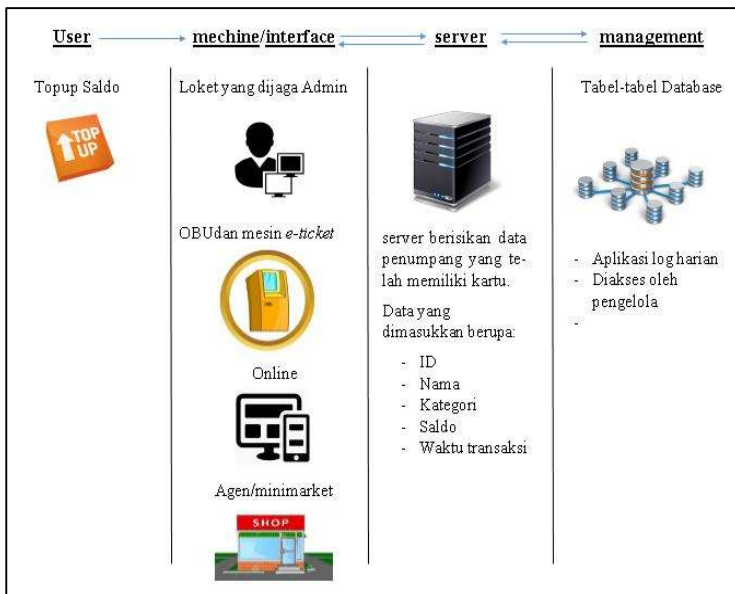
Pembayaran perjalanan dilakukan untuk pembayaran kegiatan transportasi menggunakan AMC Surabaya. Pembayaran hanya dapat dilakukan melalui mesin OBU yang akan disediakan pada tiap AMC Surabaya melalui kartu yang dipegang oleh masing – masing penumpang. Kartu yang dipegang penumpang hanya akan berisi ID yang dapat digunakan untuk mengakses akun dari pemegang kartu sehingga ketika melakukan pembayaran penumpang hanya perlu menempelkan kartu pada mesin OBU dan saldo yang terdapat pada akun tersebut akan otomatis terpotong. Pada mesin OBU akan diberikan aplikasi menggunakan aplikasi Borland Delphi 6 untuk pemotongan saldo langsung pada kartu RFID yang dibawa oleh penumpang. Pada tarif yang dikenakan akan disesuaikan dengan kategori kartu yang digunakan masing-masing penumpang.

Top-up saldo kartu penumpang merupakan kegiatan pengisian ulang saldo kartu yang dimiliki penumpang. Saldo kartu penumpang akan otomatis terpotong bila sering digunakan penumpang untuk

bepergian menggunakan transportasi AMC Surabaya, maka dari itu diperlukan top-up saldo untuk dapat mengisi kembali saldo penumpang yang telah habis. Pengisian dapat dilakukan dengan beberapa cara. Pertama dapat dilakukan pada melalui admin yang berjaga di halte-halte tertentu. Kedua, dapat pula dilakukan melalui mesin yang diletakkan pada halte. Ketiga, apabila penumpang sudah terlanjur berada pada transportasi cerdas maka top-up dapat dilakukan langsung melalui OBU. Keempat melalui agen-agen yang bekerjasama dengan AMC Surabaya dan terakhir disediakan website dimana penumpang dapat melakukan isi ulang secara online melalui smartphone masing dengan website yang telah disediakan.

3.2 Perancangan Desain Sistem Top-Up Saldo

Secara umum akan memiliki alur kerja seperti pada gambar 3.2 berikut :



Gambar 0.2 Alur Sistem Top-Up Saldo

Desain sistem top-up saldo digunakan untuk melayani kebutuhan user /penumpang untuk melakukan pengisian ulang saldo

kartu yang dimiliki penumpang. Mesin/Interface yang dibangun melalui dua media yaitu Borland Delphi 6 yang telah terinstall pada mesin OBU dan menggunakan media website untuk dapat digunakan diluar mesin OBU. Dalam mendesain sistem top-up saldo juga membutuhkan server yang digunakan untuk merekam data transaksi dan tabel-tabel database untuk menyimpan data secara detail sesuai waktu transaksi.

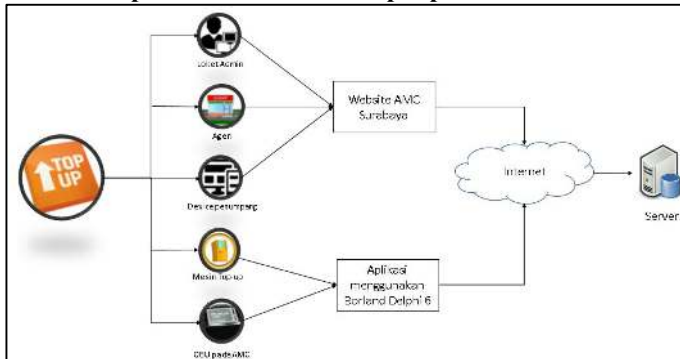
Desain sistem top-up saldo yang dirancang membutuhkan beberapa syarat untuk dapat dijalankan yaitu :

- *Applicability*, yaitu sistem dapat diterima oleh user.
- *Easy to use*, yaitu dapat digunakan oleh siapapun.
- *Reliability*, yaitu sistem handal dalam menjalankan kebutuhan user.
- *Interoperability*, yaitu dapat dioperasikan banyak penyedia layanan.
- *Efficiency*, yaitu dapat digunakan dengan efisien dengan hasil yang maksimal.
- Sistem dapat membaca kartu dengan benar
- Dapat melakukan transaksi topup saldo baik melalui aplikasi maupun website
- Delay pada aplikasi yang diterapkan pada OBU maupun mesin top-up tidak lebih dari 150ms (standar delay ITU-T, excellent)
- Seluruh proses transaksi dapat terekam dengan benar pada database server
- Dapat selalu diakses oleh admin pada Control Center Room secara real time

Dalam merancang desain sistem top-up saldo penumpang maka diperlukan pula melakukan penetapan beberapa hal dalam desain ini yaitu :

1. Penetapan metode sistem top-up saldo
2. Penetapan tarif top-up saldo kartu
3. Metode pembayaran top-up saldo
4. Struktur sistem jaringan komunikasi
5. Alur perencanaan sistem komunikasi

3.2.1 Penetapan Metode Sistem Top-Up Saldo



Gambar 0.3 Metode Sistem Top-Up Saldo

Dalam menentukan metode Topup saldo dilakukan kajian melalui penelitian pada negara0negara maju yang sudah terlebih dahulu menggunakan sistem Intelligent Transport System (ITS). Dari kajian tersebut dan penyesuaian dengan Kota Surabaya maka dapat diterapkan lima cara berbeda untuk metode top-up saldo kartu, yaitu Langsung menemui loket-loket yang dijaga petugas di halte tertentu yang tersedia mesin top-up.

1. Langsung mendatangi loket-loket yang dijaga petugas di halte yang tersedia.
2. Menggunakan mesin top-up yang disediakan di halte dengan pelayanan mandiri oleh penumpang.
3. Secara online dengan membuka website yang telah disediakan oleh pihak AMC Surabaya
4. Melalui agen-agen tertentu yang te;ah bekerjasama dengan pihak AMC Surabaya
Bila dalam keadaan terdesak dapat dilakukan dengan mesin OBU yang terdapat pada armada bus.
5. Bila dalam keadaan terdesak dapat dilakukan dengan mesin OBU yang terdapat pada armada bus.

Dari kelimat cara diatas maka terdapat dua sistem berbeda yang akan diterapkan. Pertama yaitu pada mesin top-up yang disediakan pada halte dan pada armada akan menggunakan OBU dengan bantuan

Borland delphi 6 untuk melakukan pengisian saldo dan untuk loket yang dijaga petugas, melalui sistem online menggunakan website yang telah disediakan pihak AMC Surabaya.

3.2.2 Penetapan Tarif Top-Up Saldo Kartu

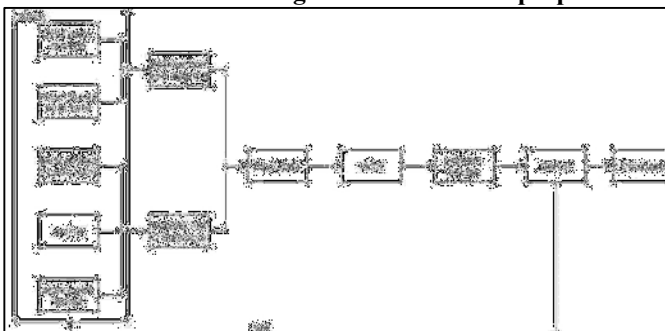
Penetapan tarif top-up saldo kartu disesuaikan dengan nominal yang akan diterapkan pada sistem. Nominal yang ditentukan pada sistem ini adalah Rp 5.000, Rp 10.000, Rp 20.000, Rp 25.000, Rp 50.000, dan Rp 100.000. Pada tarif top-up saldo tidak dikenai biaya tambahan karena pendapatan yang akan diterima perusahaan dibebankan pada tarif perjalanan.

Penetapan besar nominal yang diberikan disesuaikan dengan besaran yang biasa digunakan untuk melakukan top-up saldo pada beberapa aplikasi online yang telah marak di Indonesia. Hal ini guna memudahkan penumpang dalam hal membayar.

3.2.3 Metode Pembayaran Top-up Saldo

Untuk sistem pembayaran top-up saldo akan dapat dilakukan melalui tunai dan transfer melalui ATM. Pada loket yang dijaga petugas, mesin OBU yang terdapat pada armada, mesin top-up saldo pada halte, dan melalui agen pembayaran hanya dapat dilakukan secara tunai. Khusus untuk metode top-up secara online, pembayaran dapat dilakukan secara tunai dengan perantara agen atau loket, ataupun melalui transfer dengan ATM.

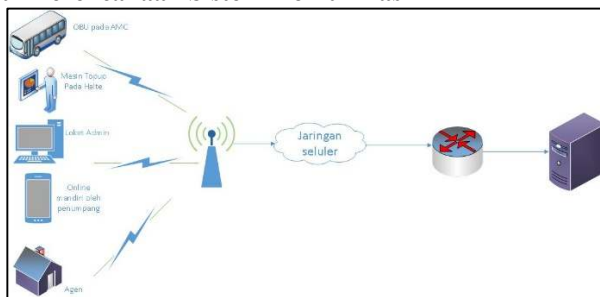
3.2.4 Struktur Sistem Jaringan Komunikasi Top-up Saldo



Gambar 0.4 Diagram Blok Sistem Jaringan Komunikasi Top-Up Saldo

Sistem jaringan komunikasi top-up saldo dapat dilihat pada blok diagram Gambar 3.4. Beberapa metode pengisian saldo kartu penumpang (client) tetap memiliki mekanisme sistem komunikasi yang sama. Pada OBU yang dipasangkan pada armada dan mesin topup yang terdapat pada halte akan diinstall aplikasi yang telah dibuat menggunakan aplikasi Borland Delphi 6. Sedangkan pada loket petugas, agen maupun online akan disediakan website yang dibuat menggunakan bahasa PHP. Pengiriman data ke server oleh client akan dilakukan setelah memasukkan data yang diperlukan sesuai dengan kolom yang tertera pada aplikasi maupun website. Pengiriman dari media-media tersebut dilakukan melalui jaringan seluler untuk sampai pada server. Setelah data diterima server maka akan terdapat reply untuk dikirim kembali pada client berupa data identitas kartu. Kemudian untuk topup maka dibutuhkan masukan data lagi berupa nominal saldo yang dimasukkan dan dilakukan pengiriman data kembali pada server dan masukan data kembali pada database. Selanjutnya, server akan kembali memberikan reply status akhir dari data penumpang tersebut.

3.2.5 Alur Perencanaan Sistem Komunikasi



Gambar 0.5 Diagram Alur Komunikasi Jaringan Terintegrasi Sistem Topup

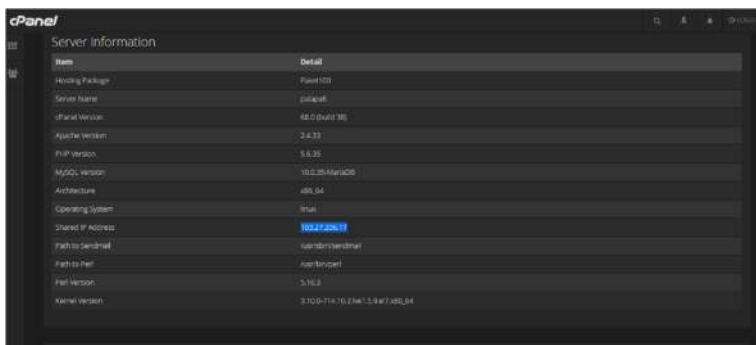
Bagian ini merupakan awal dari sistem yang dirancang, maka diperlukan tahapan perencanaan dan implementasi sistem. Perencanaan digunakan untuk mempermudah pekerjaan dari implementasi sistem yang dirancang sehingga analisis terhadap implementasi dapat dipahami, maka dibuatlah diagram alur komunikasi terintegrasi untuk top-up saldo penumpang.

Gambar 3.5 menjelaskan berlangsungnya komunikasi dari beberapa media pengisian saldo sampai pada server. Komunikasi tersebut akan mengirimkan data mengenai identitas kartu sesuai dengan ID yang terdapat pada database. Dari data tersebut maka dapat dilakukan komunikasi kembali untuk melakukan pengisian jumlah saldo sehingga pembaruan data saldo penumpang dapat dilakukan dan akan terdapat log pengisian saldo pada server database.

3.3 Perangkat Server yang Digunakan

Server yang digunakan untuk *merecord* data yang didapat adalah *b301.lawanghosting.pw/cpanel*, di mana di dalamnya terdapat *database* keseluruhan yang menyimpan data-data yang didapatkan dari setiap armada yang beroperasi. Dari data yang didapat, *Server* akan otomatis mengirimkan kembali informasi yang didapat menuju halte ataupun armada untuk memberitahukan jumlah saldo calon penumpang.

Berikut rincian informasi dari perangkat server



Item	Detail
Hosting Package	Panel 101
Server Name	idagaput
cPanel Version	60.0 (Multi OS)
Apache Version	2.4.18
PHP Version	5.6.35
MySQL Version	5.5.53-MariaDB
Architecture	x86_64
Operating System	Ubuntu
Shared IP Address	103.27.206.19
Path to Sendmail	/usr/sbin/sendmail
Path to Perl	/usr/bin/perl
Perl Version	5.16.3
Kernel Version	3.12.0-114-10.2.Ubuntu SMP x86_64

Gambar 0.6 Informasi Server

Persyaratan dalam membangun server antara lain :

1. Dapat mengatur semua aktifitas transaksi
2. Dapat memunculkan data penumpang sesuai ID yang tersimpan pada database
3. Dapat melakukan banyak permintaan dari user tanpa gangguan

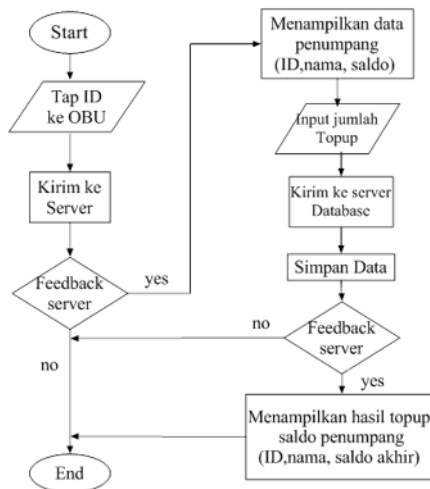
3.4 Pembuatan Aplikasi Sistem Top-Up pada OBU

Dalam pembuatan aplikasi sistem top-up pada OBU diperlukan terlebih dahulu rancangan pembuatan agar aplikasi yang akan dibuat sesuai dengan apa yang diperlukan user. Selain itu langkah – langkah pengerjaan akan ditampilkan pula untuk dapat merealisasikan hasil dari rancangan pembuatan sistem top-up.

3.4.1 Rancangan Pembuatan Aplikasi Sistem Top-Up

Dalam merancang aplikasi top-up saldo digunakan Borland Delphi 6 dikarenakan menyediakan banyak fitur yang dapat digunakan untuk membuat suatu aplikasi. Aplikasi ini akan diinstall di *OBU (On Board Unit)* yang ditempatkan pada setiap bus yang beroperasi dan pada mesin top-up saldo yang terdapat pada halte. Untuk Top-up langsung pada OBU hanya akan dilakukan apabila terdapat penumpang yang lupa mengisi saldonya terlebih dahulu sebelum menaiki AMC Surabaya dan pada mesin top-up penumpang diharuskan untuk melakukan top-up sendiri tanpa bantuan petugas.

Berikut adalah diagram alur pembuatan aplikasi pengiriman data yang dijalankan oleh OBU atau mesin top-up menggunakan aplikasi yang berasal dari borland delphi 6:



Gambar 0.7 Diagram Alur Pembuatan Sistem Top-Up

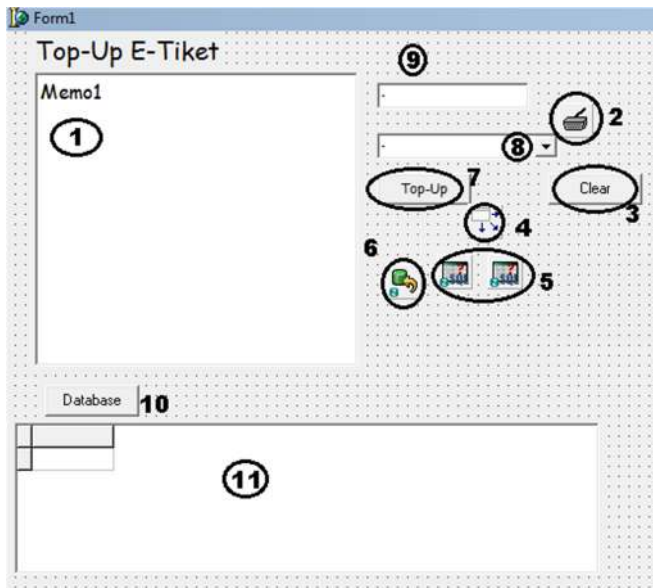
Dalam perancangan aplikasi dengan Borland Delphi 6 ini persyaratan sebagai berikut :

1. Dapat digunakan dengan mudah dengan user
2. Sistem bekerja dengan baik dan handal
3. Dapat memilih nominal top-up sesuai kebutuhan
4. Seluruh proses dapat masuk dan terekam pada server database
5. Delay penggunaan per user dan pengiriman data tidak lama
6. Throughput yang tinggi.

3.4.2 Langkah-Langkah Pembuatan Aplikasi Sistem Top-Up

Untuk membuat aplikasi sistem top-up pada OBU digunakan bantuan software Borland Delphi 6. Langkah –langkah yang harus dilakukan adalah :

1. Memilih objek-objek yang diperlukan untuk membuat tampilan muka aplikasi. Dalam pembuatan ini tampilan muka aplikasi didesain sesuai gambar 3.8.



Gambar 0.8 Rancangan Aplikasi Sistem Top-Up

2. Menuliskan variabel awal untuk skrip perintah pembuatan aplikasi top-up. Deskripsi variabel yang digunakan sebagai berikut:

```
var  
  Str,s1,ID,sKat: String;  
  SAldoAwal,xTarif:Double;
```

3. Awal program tuliskan perintah agar Memo selalu bersih tanpa ada tulisan apapun. Kemudian masukkan skrip untuk perintah membaca kartu RFID pada mesin OBU dan ditampilkan ID kartu pada Memo dan juga pada kolom Edit1.

```
begin  
  Memo1.Clear ;  
  if COut>=16 then  
    begin  
      ComPort1.ReadStr(Str, Count);  
      Memo1.Lines.Add(Str);  
      ID:=Str;  
      Edit1.Text:=Copy(Str,2,12);
```

4. Tulis perintah untuk mengambil data kategori dan saldo pada database tabel db_rfid sesuai dengan ID yang terbaca dan tempatkan pada Query1 dengan variabel sKat untuk kategori dan variabel SaldoAwal untuk jumlah saldo yang tersisa dengan format integer karena berupa angka.

```
ZQuery1.SQL.Text:=  
  'SELECT * FROM db_rfid WHERE id="'+Edit1.Text+'";  
ZQuery1.Close;  
ZQuery1.Open;  
  
  if ZQuery1.RecordCount>0 then  
  begin  
    if ZQuery1.FieldByName('Kategori').Text='Pelajar' then  
    begin  
      sKat:='Pelajar';
```

```

end
else
if ZQuery1.FieldByName('Kategori').Text='Dewasa' then
begin
sKat:='Dewasa';

end
else
if ZQuery1.FieldByName('Kategori').Text='Anak-Anak' then
begin
sKat:='Anak-anak';
end
else
if ZQuery1.FieldByName('Kategori').Text='Lansia' then
begin
sKat:='Lansia';
end;
if ZQuery1.RecordCount>0 then
begin
SaldoAwal:=ZQuery1.FieldByName('Saldo').AsInteger;

```

5. Tuliskan perintah untuk menampilkan data pada Query1 pada Memo1 mulai dari nama pemilik kartu, kategori, saldo, dan perintah untuk melakukan top-up saldo. Apabila kartu RFID ketika dibaca belum terdaftar pada database maka akan muncul pemberitahuan bahwa “Kartu belum terdaftar !”.

```

Memo1.Lines.Add(
'==>-Ini adalah kartunya'+
ZQuery1.FieldByName('Nama').Text+
#13#10' - Kategori '+sKat+
#13#10' - Saldo Awal Rp.
'+FormatFloat(',0',SaldoAwal)+
#13#10' - Silahkan melakukan Top-Up'+
#13#10");
end
else
Memo1.Lines.Add('Kartu belum terdaftar !');
end;
end;

```

6. Selanjutnya berikan perintah untuk tombol Top-up dengan mendeklarasikan kembali variabel yang digunakan.

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
Str,s1,ID,sCatatan,sKat: String;
```

```
SAldoAwal,xTarif:Double;
```

7. Berikan deklarasi untuk ComboBox agar muncul pilihan nominal dalam mengisi saldo. Pilihan yang digunakan adalah 5000, 10000, 20000, 25000, 50000, dan 100000.

```
begin
```

```
  if ZQuery1.RecordCount>0 then
```

```
  begin
```

```
    if ComboBox1.ItemIndex=0 then
```

```
    begin
```

```
      xTarif:=5000;
```

```
    end
```

```
    else if ComboBox1.ItemIndex=1 then
```

```
    begin
```

```
      xTarif:=10000;
```

```
    end
```

```
      else if ComboBox1.ItemIndex=2 then
```

```
    begin
```

```
      xTarif:=20000;
```

```
    end
```

```
      else if ComboBox1.ItemIndex=3 then
```

```
    begin
```

```
      xTarif:=25000;
```

```
    end
```

```
      else if ComboBox1.ItemIndex=4 then
```

```
    begin
```

```

xTarif:=50000;
end
    else if ComboBox1.ItemIndex=5 then
    begin
    xTarif:=100000;

    end;

```

8. Selanjutnya tuliskan perintah untuk menampilkan saldo setelah ditambahkan dengan jumlah nominal yang dipilih untuk diisi ulang.

```

SAldoAwal:=ZQuery1.FieldByName('Saldo').AsInteger;
Memo1.Lines.Add(
'- Top-up Rp. '+FormatFloat('0',xTarif)+
#13#10' - Saldo Akhir Rp. '+FormatFloat
('0',SaldoAwal+xTarif)+

#13#10");

```

9. Tuliskan perintah pembaharuan data saldo pada *database* disesuaikan dengan ID kartu dan berikan respon tulisan berhasil pada Memo.

```

Zquery2.SQL.Text:=
'UPDATE db_rfid SET Saldo=Saldo+'+FLoatToStr(xTarif)+
' WHERE ID="'+Edit1.Text+'";
ZQuery2.ExecSQL;
Memo1.Lines.Add('====>Top-Up berhasil !');

```

10. Berikan perintah yang dapat digunakan untuk memasukkan rekaman data transaksi top-up pada tabel *database* tb_logtopup.

```

sCatatan:=FLoatToStr(xTarif);
Zquery2.SQL.Text:=
'INSERT INTO tb_logtopup(CardID,Topup)
VALUES("'" +Edit1.Text+"'",sCatatan+'");
ZQuery2.ExecSQL;

```

11. Tuliskan perintah kembali untuk tombol database agar tombol tersebut dapat digunakan untuk membuka data transaksi pada Query1 dan ditempatkan pada DB Grid.

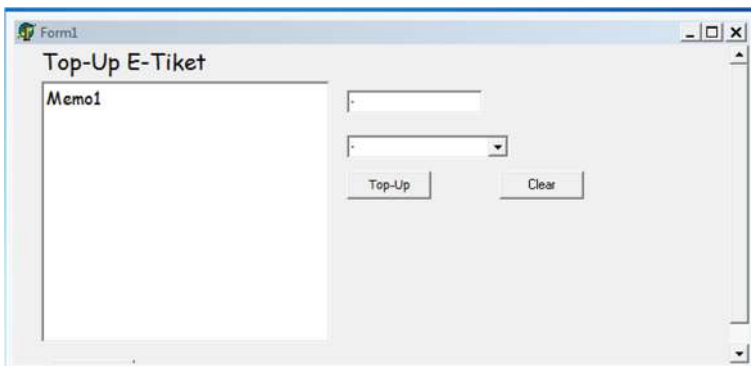
```
procedure TForm1.Button2Click(Sender: TObject);  
begin  
    ZQuery1.Close;  
    ZQuery1.Open;  
end;
```

12. Tuliskan perintah juga pada tombol clear yang dapat difungsikan untuk menghapus tulisan apapun yang terdapat pada memo.

```
procedure TForm1.Button3Click(Sender: TObject);  
begin  
    Memo1.clear;  
end;
```

13. Ubah dan berikan tambahan informasi letak dan nama database pada objek ZConnection agar ketika aplikasi berjalan dapat langsung dihubungkan dengan server.

14. Tampilan aplikasi top-up saldo pada OBU yang telah dibuat dapat dilihat pada gambar berikut :



Gambar 0.9 Tampilan Aplikasi Top-Up Sistem pada OBU

3.5 Pembuatan Aplikasi Sistem Top-Up Berbasis Web

Sesuai rancangan penetapan metode sistem top-up saldo maka diperlukan aplikasi lain untuk dapat diterapkan diluar mesin OBU dan

mesin top-up. Untuk hal tersebut maka dibuatlah aplikasi sistem top-up berbasis web. Dalam pembuatannya maka perlu rancangan untuk dapat mempermudah dalam langkah-langkah pembuatan aplikasi ini.

3.5.1 Rancangan Aplikasi Sistem Top-Up

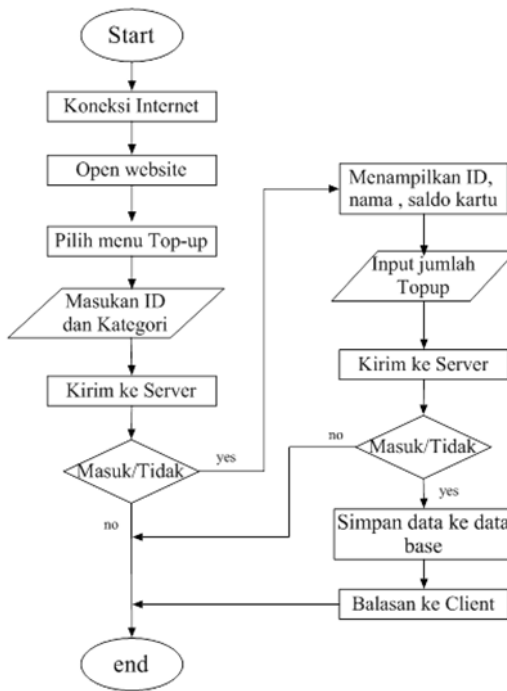
Dalam membuat rancangan aplikasi sistem top-up berbasis web dipilih program membuat bahasa PHP karena bahasa ini mudah digunakan dan minim koding namun tetap dapat digunakan membuat web secara maksimal. Web ini secara umum dapat mencakup kebutuhan dasar dalam mendaftarkan kartu baru, top-up dan menampilkan tabel pemilik kartu RFID. Pada fiturnya dapat disesuaikan kembali sesuai kebutuhan. Misalkan untuk Admin maka seluruh fitur dapat digunakan, sedangkan untuk penumpang hanya dapat dilakukan cek saldo dan top-up saldo saja.

Website untuk perancangan sistem ini dapat diakses melalui alamat *b301.lawanghosting.pw/bussurabaya/index.php*. Dalam membuat program untuk website ini dibagi menjadi beberapa list program. Dimulai dari program untuk membuat halaman awal yang berisikan menu-menu untuk daftar kartu baru, program untuk mengoneksikan website dengan database, program halaman untuk cek sald dengan ID, prgram proses cek saldo, program halaman top-up saldo, program untuk melakukan aktifitas topup saldo, program untuk halaman daftar kartu baru, program untuk memasukkan anggota baru pada *database server*, dan program untuk menampilkan siapa saja yang telah terdaftar pada *database server*.

Sistem website ini memiliki persyaratan sistem sebagai berikut :

1. Dapat diterima user
2. Dapat diakses segala device
3. Mudah digunakan semua kalangan
4. Sistem bekerja dengan baik dan handal
5. Efisien dalam waktu pengisian saldo maupun perekaman transaksi topup
6. Dapat memilih nominal sesuai kebutuhan

Berikut adalah diagram alur pembuatan aplikasi top-up saldo berbasis web sebagai berikut :



Gambar 0.10 Diagram Alur Pembuatan Sistem Top-Up

Yang dilakukan pertama kali adalah yang pasti *user* wajib terkoneksi dengan jaringan internet. Setelah itu, *user* membuka laman yang telah disediakan untuk pengisian saldo kartu. Setelah masuk laman, maka pilih menu Top-up dilanjutkan dengan pengisian ID kartu dan kategori kartu untuk pengecekan ID, nama, dan saldo. Setelah itu pilih nominal top-up yang telah disediakan.

Setelah dipilih jumlah nominal yang diinginkan penumpang mengklik tombol top-up maka akan terkirim request pada *server* untuk updating data saldo penumpang tersebut pada *database* yang sudah dibuat sebelumnya.

3.5.2 Langkah – Langkah Pembuatan Aplikasi Sistem Top-up

Dalam realisasi pembuatan aplikasi top-up saldo berbasis web dijelaskan sebagai berikut :

1. Buat program koneksi agar web dan database server dapat terhubung.

```
<?php

$server = "localhost";
$user = "root";
$password = "";
$nama_database = "tugasakhir";

$db = mysqli_connect($server, $user, $password,
$nama_database);

if( !$db ){
    die("Gagal terhubung dengan database: " .
mysqli_connect_error());
}

?>
```

2. Tulis program untuk halaman awal dari web. Sesuai dengan rancangan terdapat tiga menu utama untuk dapat melakukan tindakan berbeda melalui website tersebut yaitu tambah kartu baru, top-up, dan menampilkan data pemilik kartu.

```
<!DOCTYPE html>
<html>
<head>
    <title>Intelligent Transport System | Bus
Surabaya</title>
</head>
<body>
    <header>
        <h3>Intelligent Transport System Surabaya</h3>
        <h1>Bus Surabaya</h1>
    </header>

    <h4>Menu</h4>
    <nav>
```

```

        <ul>
        <li><a href="form_tambah.php">Daftar
        Baru</a></li>
        <li><a href="form_topup.php">Top-up</a></li>
        <li><a href="tampilkan.php">Kontak
        Penumpang</a></li>
        </ul>
    </nav>
    <?php if(isset($_GET['cek'])): ?>
    <p>
        <?php
            if($_GET['cek'] == 'belumterdaftar'){
                echo "kartu tidak terdaftar!";
            } else {
                echo "";
            }
        ?>
    </p>
    <?php endif; ?> </body>

```

3. Selanjutnya buat program laman ketika menu top-up diklik

```

<!DOCTYPE html>
<html>
<head>
    <title>Top-Up Saldo Penumpang | Intelligent Transport
    System Surabaya</title>
</head>

<body>
    <header>
        <h3>Top-up Saldo Penumpang</h3>
    </header>

    <form action="proses_cektopup.php" method="POST">

        <fieldset>
        <p>

```

```

<label for="id">ID: </label>

    <input name="id"></>
</p>
<p>
    <label for="kategori">Kategori: </label>
    <select name="kategori">
        <option>Anak-anak</option>
        <option>Dewasa</option>
        <option>Lansia</option>
        <option>Pelajar</option>
    </select>
</p>
<p>
    <label for="tanggal">Tanggal: </label>
    <output type="integer" name="tanggal" />
<?php
    $tgl=date('Y-m-d');
    echo $tgl;
?>
</p>
<p>
    <input type="submit" value="Cek" name="cek" />
</p>

</fieldset>

</form>

</body>
</html>

```

4. Buat program perintah untuk tombol cek saldo setelah laman top-up terbuka.

```

<?php
include("koneksi.php");
if(isset($_POST['cek'])){
    $id=$_POST['id'];
    $kategori=$_POST['kategori'];

```

```

$sql= "SELECT * FROM db_rfid WHERE id='$id'
AND kategori='$kategori'";
$query = mysqli_query($db, $sql);
if(mysqli_num_rows($query)>0) {
    $row_akun= mysqli_fetch_array($query);
    $_SESSION['id'] = $row_akun['id'];
    $_SESSION['Nama'] = $row_akun['Nama'];
    $_SESSION['Kategori'] = $row_akun['Kategori'];
    $_SESSION['Saldo'] = $row_akun['Saldo'];
} else {
die("Data tidak ditemukan");
}
}
else {
die("Data tidak ditemukan");
}
?>

```

5. Selanjutnya buat program laman untuk top-up saldo setelah pengecekan terferifikasi

```

<!DOCTYPE html>
<html>
<head>
    <title>Top-up</title>
</head>

<body>
    <header>
        <h3>Top-up Saldo Penumpang</h3>
    </header>

    <form action="proses_testopup.php" method="POST">
        <fieldset>

            <label for="id">ID: </label>
            <input type="char" name="id" value="<?php echo
$row_akun['id'] ?>" /> />

        <p>

```

```

        <label for="nama">Nama: </label>
        <input type="text" name="nama" value="<?php
echo $row_akun['Nama'] ?>" />
    </p>
    <p>
        <label for="kategori">Kategori: </label>
        <?php echo $row_akun['Kategori']; ?>
        <select name="kategori">
            <option <?php echo ($kategori == 'Anak-anak')
? "selected": "" ?>>Anak-anak</option>
            <option <?php echo ($kategori == 'Dewasa') ?
"selected": "" ?>>Dewasa</option>
            <option <?php echo ($kategori == 'Lansia') ?
"selected": "" ?>>Lansia</option>
            <option <?php echo ($kategori == 'Pelajar') ?
"selected": "" ?>>Pelajar</option>
        </select>
    </p>
    <p>
        <label for="saldo">Saldo: </label>
        <input type="integer" name="saldo"
value="<?php echo $row_akun['Saldo'] ?>" />
    </p>
    <p>
        <label for="topup">Topup: </label>
        <select name="topup">
            <option>5000</option>
            <option>10000</option>
            <option>20000</option>
            <option>25000</option>
            <option>50000</option>
            <option>100000</option>
        </select>
    </p>
    <p>
        <input type="submit" value="Top-up"
name="topklik" />
    </p>
</fieldset>

```

```

</form>
</body>
</html>

```

6. Selanjutnya buat perintah untuk proses top-up setelah tombol to-up terpilih

```

<?php
include("koneksi.php");

if(isset($_POST['topklik'])){
    $id=$_POST['id'];
    $nominal=$_POST['topup'];
    $saldo=$_POST['saldo'];
    (int)$saldoakhir=(int)$nominal+(int)$saldo;

    // buat query
    $sql = "UPDATE db_rfid SET Saldo='$saldoakhir'
    WHERE id='$id'";
    $query = mysqli_query($db, $sql);

    $sql = "INSERT INTO tb_logtopup (CardID, Topup)
    VALUE ('$id', '$nominal')";
    $query = mysqli_query($db, $sql);

    if( $query ) {
        // kalau berhasil alihkan ke halaman index.php
        dengan status=sukses
        header('Location: topupsukses.php?topup=sukses');
    } else {
        // kalau gagal alihkan ke halaman indek.php dengan
        status=gagal
        header('Location: topupsukses.php?topup=gagal');
    }
}

else {
    die("Topup gagal");
}
?>

```

7. Tampilan website yang telah dibuat



Gambar 0.11 Tampilan Awal Website

3.6 Perancangan Database PHPMyAdmin dan Report System

Server yang digunakan untuk merekam data yang didapat adalah *b301.lawanghosting.pw* dengan memanfaatkan database PHPMyAdmin pada *Server*. Data yang dikirimkan untuk permintaan top-up saldo ke *Server* akan disimpan didalam database *b301lawa putri*. Berikut penjelasan masing-masing kolom yang berada di database:



Gambar 0.12 Database Top-Up Sistem

Pada gambar diatas menunjukkan tabel-tabel yang berada pada *database server*. Tabel-tabel itu antara lain :

- 1) *db_rfid* yang berisikan data- data penumpang pemilik kartu RFID. Terdapat ID, nama, kategori, saldo, dan waktu terakhir melakukan transaksi
- 2) *tb_logrfid* berisikan data *log* pembayaran yang dilakukan pada tiap pembayaran e-tiket menggunakan OBU

- 3) `tb_logtopup` berisikan data *log* pengisian saldo kartu penumpang yang didapat dari pembelian melalui website ataupun melalui OBU.

Selain *database* yang sudah dirancang, media penyimpanan lainnya adalah pada web report, seluruh aktivitas yang berkaitan dengan keseluruhan manajemen revenue akan ditempatkan di dalamnya, seperti log aktifitas pembayaran, dan log topup saldo. Semua bisa dicek di `b3011.lawanghosting.pw/report/index.php`



Gambar 0.13 Laman webreport ITS

3.7 Skenario Pengujian Sistem

Setelah proses implementasi sistem telah selesai, langkah selanjutnya adalah menentukan skenario pengujian sistem, apa saja yang akan di uji dan apakah pengujian tersebut sudah sesuai dengan implementasi yang dijelaskan sebelumnya atau belum. Pengujian sistem yang di rancang meliputi pengujian sistem *client-server* serta *fitur* lainnya yang ada di dalam OBU/website dan pengujian jaringan. Pada pengujian jaringan, dibuat skenario pengujian QoS dari banyaknya parameter yang berkaitan dengan QoS dalam tugas akhir ini hanya mencari nilai *throughput* dan *delay* nya saja.

3.7.1 Parameter Pengujian Realisasi Sistem

Pengujian realisasi sistem top-up saldo memiliki target luaran yang beragam sesuai dengan fungsi yang diterapkan dari masing-

masing *fitur* yang ada. Berikut adalah syarat-syarat dari perancangan sistem top-up saldo penumpang.

Tabel 0.1 Target Uji

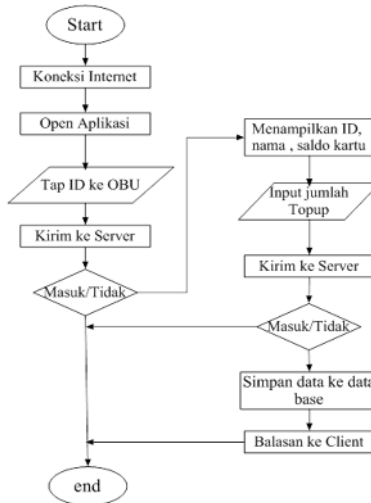
No.	Objek Uji	Fitur Uji	Tujuan Pengujian
1	OBU	Top-up saldo menggunakan aplikasi Borland Delphi 6	Untuk melihat pengiriman informasi dan pembaruan data saldo pada database.
2	Website	Top-up saldo menggunakan website	Untuk melihat pengiriman informasi dan pembaruan data saldo pada database.
3	Jaringan	Pengujian Throughput	Mengetahui nilai throughput rata-rata sistem jaringan pada aplikasi borland delphi 6 menuju server
4	Jaringan	Pengujian Delay	Mengetahui nilai Delay rata-rata sistem jaringan pada aplikasi borland delphi 6 menuju server
5.	Server	Pengujian Server	Mengetahui kinerja server ketika mendapat banyak permintaan <i>client</i>

3.7.2 Pengujian Aplikasi Top-up Saldo Menggunakan OBU

Pengujian aplikasi top-up saldo menggunakan OBU dilakukan untuk mengetahui apakah sistem telah sesuai dengan yang diharapkan atau belum. Pengujian ini dilakukan dengan cara melakukan proses pengiriman data dari *client* (mesin OBU) menuju *server* menggunakan aplikasi dari *software* Borland Delphi 6 yang telah dibuat dan diinstall pada perangkat OBU yang berada pada armada maupun mesin top-up yang ditempatkan pada halte. Proses ini dilakukan untuk mengetahui apakah data yang dikirimkan dapat diterima oleh *server* dan apakah *client* (mesin OBU) mendapatkan balasan dari *server*.

Selain itu, pengujian ini juga dilakukan untuk mengetahui apakah data yang diterima oleh server dapat disimpan pada *database server* atau tidak. Data yang dikirimkan berupa ID kartu untuk mendapatkan data penumpang yang sesuai dengan kartu dan setelah penginputan kembali nominal top-up pengiriman data kembali dilakukan

oleh client menuju server untuk selanjutnya dilakukan pembaruan data saldo dan perekaman data transaksi yang juga terdapat pada database server.



Gambar 0.14 Flowchart Pengujian Aplikasi menggunakan OBU

Alur pengujian aplikasi top-up saldo menggunakan OBU ditunjukkan pada Gambar 3.13. Adapun langkah – langkah dalam pengujian aplikasi top-up saldo adalah sebagai berikut :

Mengoneksikan perangkat OBU dengan jaringan internet dengan menggunakan jaringan seluler.

- Membuka aplikasi pada Obu dan jalankan program
- Menempelkan kartu RFID pada mesin OBU untuk pembacaan ID dan dikirimkan pada server agar mendapatkan informasi berupa nama pemilik kartu, kategori, dan saldo
- Setelah data berhasil didapat, memasukkan nominal yang diinginkan untuk penambahan saldo kartu. Klik tombol topup yang terdapat pada aplikasi maka proses top-up akan berlangsung.

- Jika pesan topup dapat diterima *server* maka akan terdapat pesan balasan apabila topup berhasil.
- Apabila data berhasil diperbarui server maka data akan tersimpan pada *database server* dan OBU akan menampilkan balasan keberhasilan. Selain itu pada *database server* sendiri akan terjadi updating data sesuai dengan pengisian yang berlangsung.

Parameter yang diukur pada pengujian ini adalah pembacaan kartu RFID pada mesin OBU menggunakan aplikasi Borland Delphi 6, pengecekan data kartu setelah pengiriman nomer ID pada server, penginputan nominal yang akan ditambahkan pada saldo dan penerimaan data oleh server untuk diperbarui, dan pembaruan serta penyimpanan data transaksi pada *database server*.

3.7.3 Pengujian Aplikasi Top-up Saldo Menggunakan Website

Pengujian ini dilakukan untuk melihat apakah sistem topup saldo yang dilakukan secara online melalui website dapat bekerja dengan baik. Langkah – langkah pengujian aplikasi sebagai berikut :

- 1) Sebelum pengujian dimulai terlebih dahulu perangkat harus terkoneksi dengan internet. Perangkat yang digunakan dapat berupa komputer atau *smartphone*, asalkan dapat membuka aplikasi browser.
- 2) Kemudian meketikkan alamat web AMC Surabaya yaitu b301.lawanghosting.pw/bussurabaya/index.php
- 3) Terdapat tiga pilihan pada website tersebut, yaitu untuk daftar kartu baru, top-up dan kontank penumpang. Untuk sistem Top-up sendiri dapat langsung dipilih menu Top-up pada baris ke dua menu.
- 4) mengisikan ID dan kategori dari kartu penumpang yang akan diisikan saldo.
- 5) Kemudian klik tombol cek pada laman tersebut. Cek ini untuk mengambil data kartu pada *database server*. Apabila data yang diisikan benar maka akan muncul tampilan top-up saldo dan bila data yang diisikan salah maka akan muncul pemberitahuan data tidak ditemukan
- 6) Setelah lama top-up terbuka, memilih nominal topup pada pilihan dilaman tersebut.

- 7) Setelah itu klik tombol top-up. Updating data pada *database server* akan dilakukan setelah itu. Bila berhasil akan muncul laman pemberitahuan seperti gambar berikut
- 8) Pada *database server* sendiri akan terjadi updating data sesuai dengan pengisian yang berlangsung. Log data yang telah melakukan topup juga akan terekam pada tabel tersendiri yang telah dirancang. *Database* untuk sistem top-up melalui web ataupun OBU berada pada tabel yang sama sehingga tidak terpisah-pisah.

Pengujian dilakukan dengan menggunakan metode *Black Box*. Parameter yang digunakan adalah pemanggilan laman website, penerimaan data sesuai inputan pengguna, penginputan nominal yang akan ditambahkan pada saldo, penerimaan data oleh server untuk pembaruan data, dan penyimpanan data transaksi pada *database server*.

3.7.4 Pengujian Implementasi Jaringan OBU dan Website

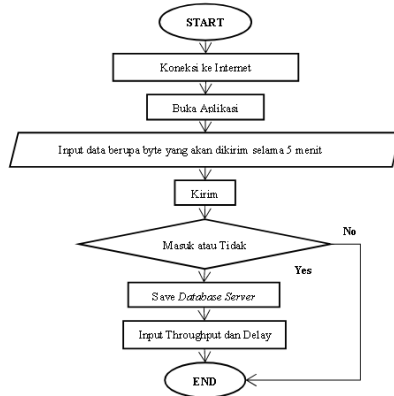
Setelah *prototype* telah selesai, dan beberapa *fitur* sudah bisa di operasikan, untuk selanjutnya pengujian implementasi jaringan komunikasi sistem dilakukan untuk mendapatkan ukuran data terbaik dalam pengiriman paket data dari *client* menuju ke *server* dalam arti pengiriman pesan dari armada yang sedang beroperasi menuju *database* atau *webreport*.

Jaringan yang dipakai pada tugas akhir yang dikerjakan ini menggunakan jaringan wireless yang di tempatkan di setiap armada ataupun halte. Terdapat sebuah modem *wireless* dengan jaringan seluler yang memiliki paket data untuk menghubungkan antara *client* menuju *server*.



Gambar 0.15 Implementasi Sistem Jaringan

Flowchat pengujian jaringan dapat digambarkan pada Gambar 3.16.



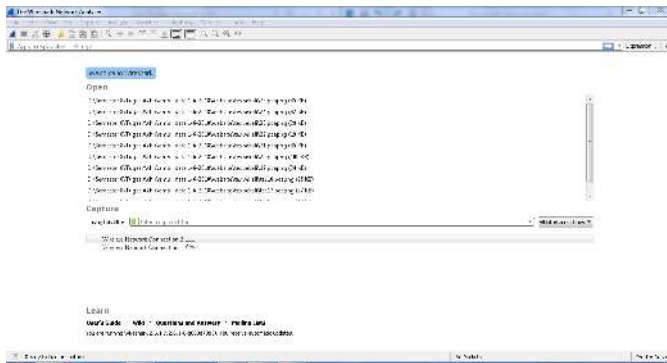
Gambar 0.16 Flowchart Pengujian Jaringan

Pengujian jaringan AMC Surabaya dilakukan dengan bantuan aplikasi wireshark. Aplikasi ini dijalankan bersamaan dengan berjalannya sistem top-up ketika menggunakan aplikasi menggunakan OBU maupun website. Aplikasi wireshark akan membantu untuk mendapatkan nilai throughput dan delay selama proses top-up saldo.

3.7.4.1 Pengujian Jaringan Menggunakan OBU

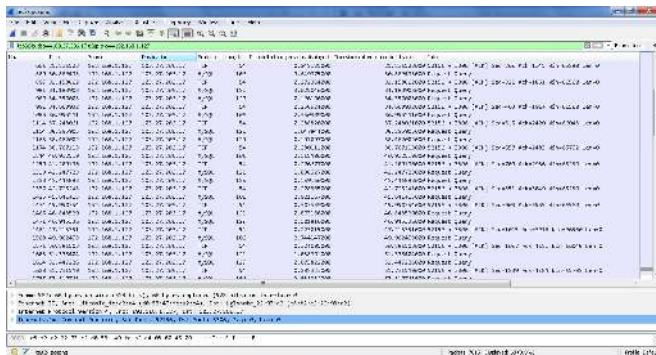
Pengujian jaringan menggunakan OBU mengambil waktu selama tiga hari masing- masing dengan tiga kurun waktu yaitu pada pagi hari (pukul 08.00), siang hari (pukul 13.00), dan sore hari (pukul 16.00). Untuk terkoneksi pada jaringan internet digunakan jaringan seluler provider telkomsel. Langkah – langkah pengujian jaringan sebagai berikut :

1. Buka aplikasi top-up pada OBU dan *software* wireshark. Pada aplikasi wireshark sebelum memulai pilih jaringan yang akan digunakan sesuai gambar berikut



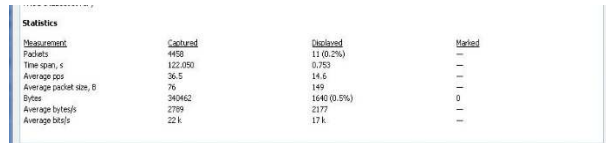
Gambar 0.17 tampilan awal wireshark

2. Menjalankan aplikasi topup pada OBU seperti biasa. Pengoprasian pada OBU dilakukan selama satu menit dan akan didapatkan data pada aplikasi wireshark.
3. Menyaring pengiriman informasi dari OBU menuju *server* untuk menampilkan hasil yang di amati saja, dengan cara mengetik “tcp && ip.dst==192.168.1.101 && ip.src==103.27.206.17” lalu menekan *enter* dan kemudian *wireshark* akan menampilkan hasil pengiriman informasi yang dikirimkan dari OBU menuju *server*.



Gambar 0.18 Tampilan pengukuran wireshark

4. Mengambil nilai delay dilihat melalui kolom waktu sejak awal frame pada panjang 54 Byte. Kemudian untuk hasil throughput dilihat pada menu statistic pada capture file properties. Tampilan yang akan muncul sebagai berikut :

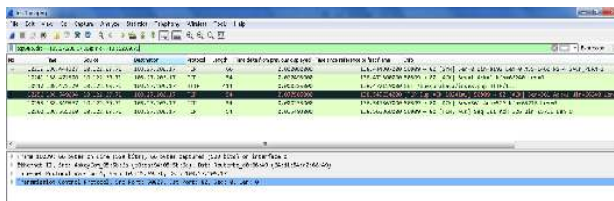


Measurement	Captured	Decoded	Matched
Packets	4458	11 (0.2%)	—
Time spent, s	122.050	0.753	—
Average gss	36.5	14.6	—
Average packet size, B	76	149	—
Bytes	340462	1648 (0.5%)	0
Average bytes/s	2789	2377	—
Average bits/s	22 k	17 k	—

Gambar 0.19 Tampilan untuk melihat nilai throughput

3.7.4.2 Pengujian Jaringan Menggunakan Website

Langkah langkah pengujian jaringan menggunakan website sama dengan pada OBU. Namun, skenario pada pengujian website hanya dilakukan satu kali dengan pengambilan sampel sebanyak 25 kali. Pengukuran dilakukan ketika membuka halaman awal dari website. Tampilan pada pengujian dapat dilihat pada gambar 3-30.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.100	192.168.1.1	HTTP	100	GET / HTTP/1.1
2	0.000000	192.168.1.1	192.168.1.100	HTTP	100	200 OK
3	0.000000	192.168.1.100	192.168.1.1	HTTP	100	GET / HTTP/1.1
4	0.000000	192.168.1.1	192.168.1.100	HTTP	100	200 OK
5	0.000000	192.168.1.100	192.168.1.1	HTTP	100	GET / HTTP/1.1
6	0.000000	192.168.1.1	192.168.1.100	HTTP	100	200 OK
7	0.000000	192.168.1.100	192.168.1.1	HTTP	100	GET / HTTP/1.1
8	0.000000	192.168.1.1	192.168.1.100	HTTP	100	200 OK
9	0.000000	192.168.1.100	192.168.1.1	HTTP	100	GET / HTTP/1.1
10	0.000000	192.168.1.1	192.168.1.100	HTTP	100	200 OK

Gambar 0.20 Cntoh Pengujian Jaringan Berbasis Web

3.7.5 Pengujian Kinerja Server

Pengujian sistem website dan server dilakukan untuk mengetahui apakah sistem telah sesuai dengan yang diharapkan atau belum. Pengujian ini dilakukan dengan menggunakan aplikasi JMeter dimana dalam skenarionya akan dilakukan beberapa kali dengan jumlah user yang berbeda beda, yaitu mulai dari 500, 750, 1000, 1250, dan 1500. Parameter yang diukur pada pengujian ini adalah error, respon time, standard deviasi, dan CPU Load server.

Pengujian kinerja server dilakukan dengan langkah-langkah :

- 1) Membuka *software* JMeter pada laptop.
- 2) Melakukan klik kanan Test Plan > Add > Threads (User) > Thread Group.
- 3) Mengisikan jumlah user yang diinginkan. Untuk tes pertama diisi dengan user 500 dengan setiap user memiliki delay 1 detik dan setiap user meminta request server sebanyak 10 kali.
- 4) Melakukan klik kanan Tread group > Add > Sampler > HTTP Request
- 5) Mengisi server name atau IP server sesuai server yang digunakan.
- 6) Untuk memantau hasil grafik, melakukan klik lagi Tread Group > Listener > Graph result
- 7) Untuk memantau hasil *respon time*, melakukan klik lagi Tread Group > Listener > Respon time graph
- 8) Untuk mengambil hasil rangkuman tes srver klik lagi Tread Group > Listener > summary report
- 9) Menunggu JMeter bekerja hingga jumlah user yang diinginkan
- 10) Mencatat hasil dari pengukuran JMeter sesuai dengan yang dibutuhkan dalam menguji server.
- 11) Mengulangi tes server dari langkah 3 dengan jumlah user yang berbeda sesuai dengan skenario pengujian server.

Halaman ini sengaja dikosongkan

BAB IV

HASIL PENGUJIAN DAN ANALISIS

Pada bab ini akan di jelaskan hasil pengujian dan analisis dari perancangan sistem komunikasi terintegrasi yang sudah dijelaskan pada bab sebelumnya, serta menjelaskan *fitur top-up* yang ada di dalam OBU juga pada website yang telah dibuat. Sistem ini menggunakan aplikasi yang di buat dengan OBU dimulai dari proses pengiriman informasi dari oleh *client* hingga dapat diterima oleh *server*. Pengujian ini di bagi menjadi beberapa bagian yaitu pada bagian aplikasi top-up pada *OBU*, website, jaringan, dan server.

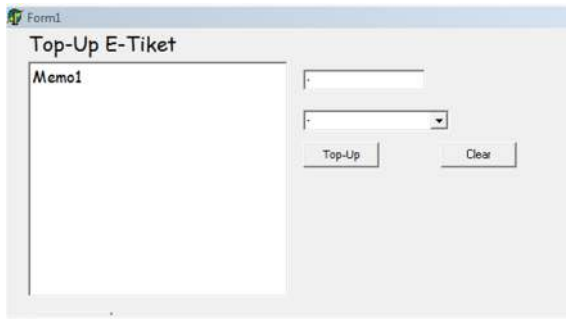
4.1 Hasil Pembuatan Aplikasi Sistem Top-Up Saldo

Pada sub ini akan diperlihatkan hasil dari pembuatan aplikasi pada OBU yang mana menggunakan bantuan software Borland Delphi 6 dan juga aplikasi top-up berbasis web.

4.1.1 Hasil Tampilan Aplikasi Top-Up pada OBU

Hasil tampilan aplikasi top-up saldo pada OBU akan diperlihatkan dalam bentuk gambar-gambar 4.1, 4.2, dan 4.3 sebagai berikut :

- a. Gambar 4.1 merupakan tampilan awal aplikasi top-up saat pertama kali aplikasi dijalankan..



Gambar 0.1 Tampilan Awal Aplikasi Top-Up

- b. Gambar 4.2 menjadi tampilan aplikasi setelah kartu RFID ditempelkan pada mesin OBU. Terdapat keterangan nama, kategori, dan jumlah saldo tersisa.

Form1

Top-Up E-Tiket

000017B61CBD

==>-Ini adalah kartunya Ilham

- Kategori Dewasa
- Saldo Awal Rp. 25,000
- Silahkan melakukan Top-Up

000017B61CBD

Top-Up Clear

5000

5000

10000

20000

25000

50000

100000

Clear

Gambar 0.2 Tampilan Aplikasi Hasil Respon Server Setelah Tapping dan Pilihan Nominal Top-up

- c. Gambar 4.3 menampilkan hasil setelah top-up berhasil.

Form1

Top-Up E-Tiket

- Kategori Dewasa
- Saldo Awal Rp. 25,000
- Silahkan melakukan Top-Up
- Top-up Rp. 5,000
- Saldo Akhir Rp. 30,000

====> Top-Up berhasil !

000017B61CBD

5000

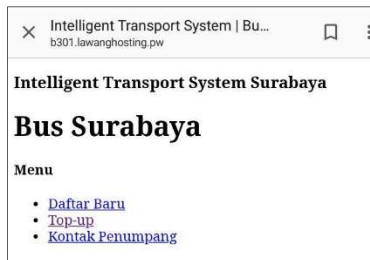
Top-Up Clear

Gambar 0.3 Tampilan Aplikasi Setelah Pengisian Saldo

4.1.2 Hasil Pembuatan Aplikasi Top-Up Berbasis WEB

Tampilan menu utama aplikasi top-up berbasis web yang telah dibuat tampak seperti pada gambar-gambar berikut ini :

- 1.) Gambar 4.4 menampilkan tampilan awal laman. Terdapat tiga menu utama dalam laman tersebut yaitu daftar baru, top-up, dan kontak penumpang




Gambar 0.4 Tampilan Halaman Awal Website

- 2.) Gambar 4.5 menunjukkan hasil dari eksekusi menekan tombol top-up. Sebelum masuk dalam laman top-up terlebih dahulu melakukan pengecekan ID dan kategori dari kartu yang dimiliki penumpang

A screenshot of a web browser showing a form titled 'Top-up Saldo Penumpang'. The browser's address bar shows 'Top-Up Saldo Penumpang | Intelli...' and 'b301.lawanghosting.pw'. The form contains three input fields: 'ID:' followed by a text box, 'Kategori:' followed by a dropdown menu currently showing 'Anak-anak', and 'Tanggal:' followed by a text box containing '2018-06-02'. At the bottom of the form is a button labeled 'Cek'.

Gambar 0.5 Tampilan Halaman Cek No. ID dan Kategori Pemilik Kartu

- 3.) Bila no. ID dan kategori benar maka laman top-up akan muncul seperti pada gambar 4.6 dan apabila ID dan kategori salah maka yang muncul adalah laman seperti pada gambar 4.7



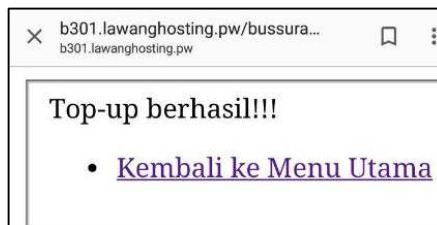
The screenshot shows a web browser window with the address bar displaying 'Top-up' and 'b301.lawanghosting.pw'. The page title is 'Top-up Saldo Penumpang'. The form contains the following fields and controls:

- ID: 0000413BFE84 (text input)
- Nama: Ufi (text input)
- Kategori: Lansia (dropdown menu with 'Lansia' selected)
- Saldo: 26000 (text input)
- Topup: 5000 (dropdown menu)
- Top-up (button)

Gambar 0.6 Tampilan Laman Top-Up Bila No. ID dan Kategori Benar



Gambar 0.7 Tampilan Laman Saat Isian No. ID dan Kategori Salah



Gambar 0.8 Tampilan Top-Up Berhasil

4.2 Hasil Pengujian Sistem Topup Berbasis OBU

Pada pengujian sistem top-up berbasis OBU dilakukan dua kali pengujian yang berbeda. Pengujian yang pertama adalah pengujian operasional sistem top-up. Pengujian ini untuk membuktikan apakah aplikasi telah bekerja sesuai rancangan atau belum. Pengujian kedua adalah pengujian performansi jaringan sistem topup dimana hal ini bertujuan untuk melihat bagaimana kinerja jaringan bila menggunakan media aplikasi top-up yang telah dirancang.

Sesuai dengan skenario pengujian yang telah dijelaskan pada bab 3, pada pengujian sistem ini akan melakukan uji aplikasi top-up saldo menggunakan OBU. Pengujian ini bertujuan untuk mengetahui apakah informasi yang dikirimkan dari armada dapat terkirimkan ke *server* dan dapat di simpan di dalam *database* atau tidak, ataupun sebaliknya informasi yang telah di terima *server* apakah bisa melakukan balasan kembali menuju armada yang sedang beroperasi atau tidak.

4.2.1 Hasil Pengujian Operasional Sistem Top-up

Hasil yang didapat dari pengujian sistem membutuhkan beberapa analisa agar dapat diketahui bila sistem berjalan dengan baik. Analisis tersebut meliputi apakah rancangan sistem *client-server* yang berjalan sesuai dengan yang telah dirancang. Berikut tabel yang mengandung analisis pengujian sistem Topup pada OBU :

Tabel 0.1 Hasil Operasional Sistem Topup

No	Pengujian	Hasil yang Didapat	Analisis
1.	Pembacaan kartu RFID pada aplikasi Delphi	Perangkat OBU berhasil membaca kartu dan tertampil pada aplikasi Delphi	Berhasil
2.	Pengecekan data kartu yang terdapat pada <i>database server</i>	<i>Server</i> berhasil memberikan data kartu sesuai yang terdapat pada <i>database server</i>	Berhasil
3.	Penginputan nominal yang akan ditambahkan pada saldo kartu dan penerimaan data oleh <i>server</i> agar diperbarui	<i>Server</i> berhasil membaca pembaruan data pada salah satu data yang diperbarui untuk ditop-up	Berhasil
4.	Pembaruan dan penyimpanan data pada <i>server database</i>	Data dapat diperbarui dan tersimpan pada <i>database server</i> PHP-MyAdmin	Berhasil

4.2.2 Hasil pengujian Performansi Jaringan Sistem Topup

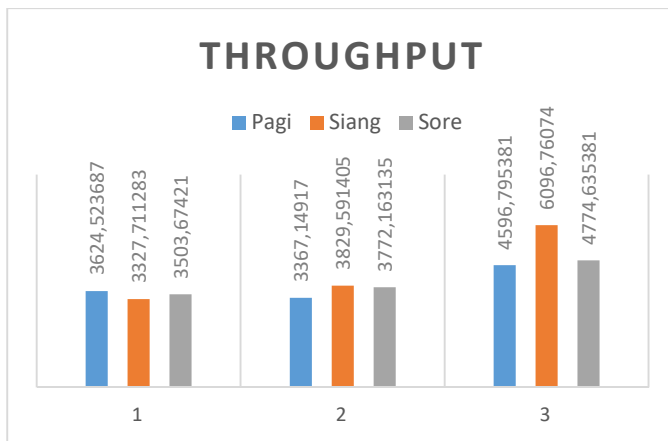
Pengujian jaringan ini dilakukan dengan cara mengirimkan data dengan kartu RFID secara terus menerus tanpa terputus selama dua menit untuk mengetahui nilai QoS jaringan seluler yang digunakan. *Software Wireshark* dijadikan alat bantu untuk dapat menemukan parameter QoS yang dibutuhkan. Parameter QoS yang dicari adalah nilai *throughput* dan *delay* yang di dapat dalam proses pengiriman informasi dari OBU yang berada di tiap bus menuju ke *server*.

Pengujian dilakukan di laboratorium AJ 404 selama 3 hari pada pagi hari pukul 08.00 dan siang hari pada pukul 13.00 dan sore pukul 16.00. Hasil dari uji jaringan ini dapat dilihat pada Lampiran C hasing pengujian jaringan dengan OBU. Terdapat sembilan data yang dihasilkan dari pengujian ini. Masing – masing data yang terkumpul sebanyak 25 sampel. Dari sampel tersebut dihitung kembali rata-rata dari tiap pengujian. Hasil perhitungan dapat diketui melalui tabel dibawah ini.

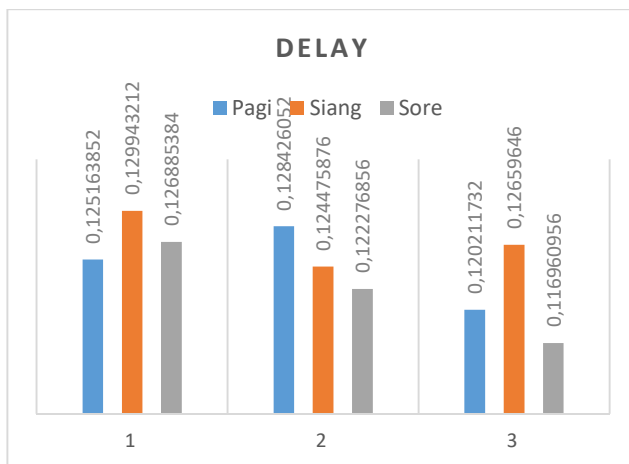
Tabel 0.2 Hasil Pengujian Jaringan

No.	Pengujian	Rata – Rata Delay (s)	Rata – Rata Throughput (bits/s)
1	Pengujian 1 (hari 1 pagi)	0,1251639	3624,52369
2	Pengujian 1 (hari 1 siang)	0,1299432	3327,71128
3	Pengujian 1 (hari 1 sore)	0,1268854	3503,67421
4	Pengujian 2 (hari 2 pagi)	0,1284261	3367,14917
5	Pengujian 2 (hari 2 siang)	0,1244759	3829,59141
6	Pengujian 2 (hari 2 sore)	0,1222769	3772,16313
7	Pengujian 3 (hari 3 pagi)	0,1202117	4596,79538
8	Pengujian 3 (hari 3 siang)	0,1265965	6096,76074
9	Pengujian 3 (hari 3 sore)	0,116961	4774,63538
	Rata-rata	0,1245489	4099,22271

Hasil yang didapatkan yaitu nilai *throughput* terbesar terjadi pada hari ke tiga waktu siang hari yaitu dengan nilai 6096,761 bits/detik dan terendah pada hari pertama waktu siang hari dengan nilai 3327,711 bits/detik. Rata-rata throughput dari semua pengujian bekisar 4099 bits/s. Grafik pengukuran dapat dilihat pada gambar berikut :



Gambar 0.9 Hasil Throughput pada OBU



Gambar 0.10 Hasil Delay pada OBU

Hasil pengujian *delay* yang telah dilakukan tersebut, dapat dianalisis bahwa nilai *delay* terbesar yaitu berada pada hari pertama waktu siang hari dimana berada dinilai 129,943 ms. Sedangkan nilai *delay* terkecil berada di hari ketiga waktu sore hari yaitu pengukuran

dimana berada di nilai 116,961 ms. Rata-rata dari pengujian diatas sebesar 124,5489 ms, dari hal ini dapat dilihat bahwa nilai delay berada pada kondisi yang normal sesuai dengan standar ITU-T *recommendation* Y1541 dimana nilai *delay* <150 ms berada pada kondisi yang *excellent*.

4.2 Hasil Pengujian Sistem Topup Berbasis Website

Pada pengujian Sistem Top-up berbasis website dilakukan dua kali pengujian yang berbeda. Pengujian pertama adalah pengujian operasional sistem top-up dan yang kedua adalah pengujian performansi jaringan sistem topup.

4.2.1 Hasil Pengujian Operasional Sistem Top-Up

Hasil yang didapat dari pengujian sistem website membutuhkan beberapa analisa agar dapat diketahui bila sistem berjalan dengan baik. Analisis tersebut meliputi apakn rancangan sistem *client-server* yang berjalan sesuai dengan yang telah dirancang. Berikut tabel yang mengandung anlisis pengujian sistem :

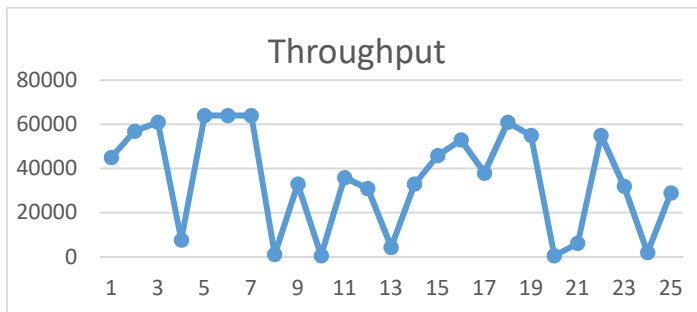
Tabel 0.3 Hasil Operasional Sistem Top Up Website

No	Pengujian	Hasil yang Didapat	Analisis
1.	Memanggil laman sistem	<i>Server</i> berhasil membukakan laman yang diinginkan	Berhasil
2.	Penerimaan data sesuai inputan <i>client</i>	<i>Server</i> berhasil memberikan data kartu sesuai yang terdapat pada <i>database server</i>	Berhasil
3.	Penginputan nominal yang akan ditambahkan pada saldo kartu dan penerimaan data oleh <i>server</i> agar diperbarui	<i>Server</i> berhasil membaca updating data pada salah satu data yang diperbarui untuk ditop-up	Berhasil
4.	Pembaharuan dan penyimpanan data pada <i>server database</i>	Data dapat diperbarui dan tersimpan pada <i>database server</i> PHPMyAdmin	Berhasil

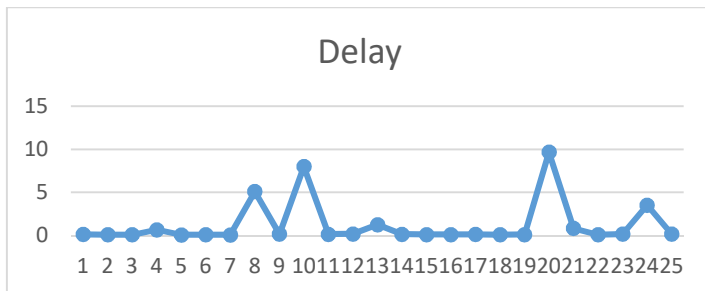
4.2.2 Hasil Pengujian Performansi Sistem Top-Up

Sesuai dengan skenario, pengujian website hanya dilakukan satu waktu dengan mengambil 25 sampel, masing – masing diukur tingkat delay dan throughputnya. Pada pengujian throughput dihasilkan throughput tersebar pada sampel 5,6, dan 7 yaitu dengan besaran 64000 bits/s dan terkecil pada sampel 20 dengan nilai 525 bits/s. Rata-rata throughput yang dihitung bernilai 35184,3 bits/s. Untuk pengujian delay didapat nilai terbesar delay pada sampel ke 20 dengan nilai 9,87 detik dan delay terkecil pada sampel ke 5 dengan nilai 0,079 detik. Rata – rata delay keseluruhan adalah 1,25s sehingga dapat dikatakan sangat bagus sesuai dengan standar ITU-T REC-G. 1010 dimana nilai delay webrowsing kurang dari 10 detik.

Hasil dari pengujian performansi sistem top-up dapat dilihat pada tabel yang terdapat pada lampiran C dan grafik dibawah ini :



Gambar 0.11 Grafik Throughput



Gambar 0.12 Grafik Delay

4.3 Pengujian Kinerja Server AMC Surabaya

Pengujian server bertujuan untuk mengetahui kemampuan server ketika banyak pengguna/user mengakses laman server AMC Surabaya. Pengujian dilakukan dengan menggunakan aplikasi JMeter dengan menggunakan user virtual untuk mengakses halaman aplikasi. Pada tabel dibawah ini akan ditunjukkan hasil pengujian server dengan simulasi user yang berbeda-beda yaitu mulai dari 500,750, 1000, 1250, dan 1500 user.

Tabel 0.4 Hasil Pengujian Server

No.	User	Error (%)	Standard Deviasi	Respon time (s)	CPU Load (%)
1.	500	0,78	1853,92	0,35	1
2.	750	0,89	1971,6	0,5	2
3.	1000	0,89	1977,3	0,45	2
4.	1250	0,83	1914,5	0,45	2
5.	1500	0,89	1968,4	0,5	2
6.	2500	0,896	2012	0,5	5
7.	4000	1,075	2198	0,65	8

Dari hasil pengujian server didapatkan pada jumlah user antara 500-4000 belum terjadi perbedaan yang signifikan pada kinerja server. Mulai dari error, standard deviasi, respon time, dan CPU Load masih rendah sehingga server ini mampu untuk menampung user hingga lebih dari 4000 user dalam penggunaannya.

BAB V

PENUTUP

5.1 Kesimpulan

Dari keseluruhan pembuatan rancangan dan pengujian sistem top-up berbasis OBU maupun berbasis web dapat ditarik kesimpulan sebagai berikut:

- d. Aplikasi top-up saldo menggunakan OBU maupun berbasis web pada uji operasional dapat berjalan (berhasil) sesuai dengan rancangan.
- e. Untuk sistem top-up berbasis OBU memiliki rata-rata delay sebesar 124,5 ms dan rata-rata throughput 4099,2 bits/s.
- f. Dari pengujian kinerja jaringan aplikasi top-up berbasis web didapatkan rata-rata delay sebesar 1,25 s dan rata-rata throughput sekitar 35184,3 bits/s.
- g. Hasil delay jaringan aplikasi top-up berbasis OBU maupun berbasis web keduanya sesuai dengan standar dari ITU-T *recommendation* Y1541 dan ITU-T REC-G. 1010 dimana nilai delay dikatakan dalam kondisi *excellent* dan sangat baik.
- h. Hasil pengujian kinerja server menunjukkan performa server berdasar rata-rata waktu tanggap, *error rate* dan CPU *load* dengan user antara 500-4000 masih dapat bekerja dengan maksimal.
- i. Dari keseluruhan hasil pengukuran parameter baik dari sisi pengujian operasional maupun performansi menunjukkan bahwa sistem top-up layak untuk digunakan dan dikembangkan lebih lanjut.

5.2 Saran

Adapun hal-hal yang masih bisa dikembangkan dari tugas akhir yang dibuat ini adalah:

- 1. Pembuatan untuk penjadwalan yang dapat diperbarui secara otomatis di setiap halte.
- 2. Pengembangan *prototype* dengan memasukkan *fitur-fitur* baru didalamnya, agar lebih mendukung kecanggihan dari *prototype* tersebut

3. Pengukuran serta penggunaan *provider* jaringan yang bervariasi, agar mendapatkan hasil yang maksimal dalam melakukan pengiriman serta penerimaan informasi dari *client-server*.

DAFTAR PUSTAKA

- [1] E. Damanik, "Perancangan Sistem Informasi Pembayaran Online Menggunakan Payment Gateway," vol. 13, no. 1. pp. 63–71, 2012.
- [2] R. O. L. Sihombing and M. Zulfin, "Analisis Kinerja Trafik Web Browser Dengan Wireshark Network Protocol Analyzer Pada Sistem Client-Server," *Singuda Ensikom*, vol. 2, pp. 96–101, 2013.
- [3] I. Teknologi, S. Nopember, I. T. S. Surabaya, J. Arif, and R. Hakim, "C i n i a," in *The 2nd Conference on Innovation and Industrial Applications (CINIA 2016) Road-map*, 2016, no. Cinia, pp. 53–58.
- [4] D. Amiegbebhor, "Intelligent Transport System (ITS) For the Management of Bus Operations in Sub-Sahara Africa: The Public Private Partnership (PPP) Approach," *Eur. J. Acad. Essays ISSN*, vol. 2, no. 5online, pp. 27–34, 2015.
- [6] C. M. Flath, S. Gottwalt, and J. P. Ilg, "A Revenue Management Approach for Efficient Electric Vehicle Charging Coordination," *2012 45th Hawaii Int. Conf. Syst. Sci.*, pp. 1888–1896, 2012.
- [7] D. W. Herdiyanto, Endroyono, and I. Pratomo, "Passenger Authentication and Payment System Using RFID Based On-Board Unit for Surabaya Mass Rapid Transportation," *Proceeding - 2016 Int. Semin. Intell. Technol. Its Appl. ISITIA 2016 Recent Trends Intell. Comput. Technol. Sustain. Energy*, pp. 305–310, 2017.
- [8] Yuliantoro, Prasetyo. "Rancang Bangun Protokol *E-Ticketing*," Institut Teknologi Sepuluh Nopember, Surabaya, 2015.
- [9] Affandi, Achmad dan tim. "Laporan Akhir Kajian Sistem dan Taknologi IT dalam Rangka Integrasi Angkutan Massal Cepat *Trunk* dan *Feeder* AMC Surabaya," Dinas Perhubungan Kota Surabaya, Surabaya, 2015.

Halaman ini sengaja dikosongkan

LAMPIRAN A

A.1 Lembar Pengesahan Proposal

Departemen Teknik Elektro Fakultas Teknologi Elektro - ITS	
TE 141599 Tugas Akhir – 4 SKS	
09 FEB 2018	
Nama Mahasiswa	: Ulfi Romadhani Rosadiana
Nomor Pokok	: 0711144000044
Bidang Studi	: Telekomunikasi Multimedia
Tugas Diberikan	: Semester Genap 2017/2018
Dosen Pembimbing 1	: Dr. Ir. Achmad Affandi, DEA
Dosen Pembimbing 2	: Ir. Gatot Kusrahardjo, MT
Judul Tugas Akhir	: Desain Sistem Top-Up Saldo Penumpang pada Intelligent Transport System Kota Surabaya <i>(Design of Passenger Balance Top-up System Intelligent Transport System for Surabaya City)</i>
Uraian Tugas Akhir Intelligent Transport System (ITS) Kota Surabaya masih memerlukan perkembangan sebelum sistem ini dapat benar-benar terealisasi, salah satunya adalah dalam sistem manajemen revenue. Hal yang perlu dikembangkan pada manajemen revenue adalah sistem top-up saldo penumpang yang akan menggunakan angkutan masal cepat Kota Surabaya seperti bus, trem, maupun monorail. Top-up saldo penumpang ini digunakan untuk mengisi ulang saldo penumpang yang terdapat pada database menggunakan bantuan <i>smartcard</i> untuk membaca ID penumpang. Proses top-up saldo dapat dilakukan dengan pembelian pada terminal dengan bantuan mesin top-up yang disediakan angkutan masal cepat Kota Surabaya ataupun melalui online dengan pembayaran secara transfer melalui ATM maupun credit card dengan ID yang telah sama dengan <i>smartcard</i> . Sistem yang dirancang harus disinkronisasi data center pada server dan data center pada OBU setelah melakukan kegiatan top-up saldo agar penumpang dapat langsung menggunakan pada angkutan masal cerdas Kota Surabaya. Pada penelitian ini juga diberikan usulan mengenai model bisnis dan teknologi yang digunakan pada sistem top-up saldo penumpang angkutan masal cerdas Kota Surabaya.	
Dosen Pembimbing I,	Dosen Pembimbing II,
	
Dr. Ir. Achmad Affandi, DEA NIP. 196510141990021001	Ir. Gatot Kusrahardjo, MT NIP. 195904281986011001
Mengetahui, Ketua Program Studi S1	Menyetujui, Kepala Laboratorium Jaringan Telekomunikasi
	
Deder C. Riawan, ST., M.Eng., Ph.D NIP. 197311192000031001	Dr. Ir. Achmad Affandi, DEA NIP. 196510141990021001

Halaman ini sengaja dikosongkan

LAMPIRAN B

HASIL PENGUJIAN JARINGAN SISTEM TOP-UP

B.1 Tabel Data Pengujian Menggunakan OBU

B.1.1 Pengujian 1 (Hari 1 Pagi)

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,045662	9460,820814
2	7	54	432	0,092819	4654,219502
3	10	54	432	0,1318885	3275,494073
4	12	54	432	0,1293292	3340,312938
5	15	54	432	0,1305967	3307,893691
6	17	54	432	0,131611	3282,400407
7	20	54	432	0,1314258	3287,025835
8	22	54	432	0,1307577	3303,820731
9	25	54	432	0,1323321	3264,514052
10	27	54	432	0,1310094	3297,473311
11	30	54	432	0,129189	3343,937951
12	32	54	432	0,13005	3321,799308
13	35	54	432	0,131152	3293,888008
14	37	54	432	0,1301886	3318,262889
15	40	54	432	0,1331638	3244,124905
16	42	54	432	0,130258	3316,494956
17	45	54	432	0,1316584	3281,218669
18	47	54	432	0,1304601	3311,357266
19	50	54	432	0,1304414	3311,83198
20	52	54	432	0,1301155	3320,127118
21	55	54	432	0,1317758	3278,295408
22	57	54	432	0,1302026	3317,906094
23	60	54	432	0,1329199	3250,077678
24	62	54	432	0,128092	3372,575961
25	63	54	432	0,1119978	3857,218624
	Rata- rata			0,1251639	3624,523687

B.1.2. Pengujian 2 (Hari 1 Siang)

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,145631	2966,401384
2	7	54	432	0,126255	3421,646667
3	10	54	432	0,1301674	3318,803326
4	12	54	432	0,1280538	3373,582041
5	15	54	432	0,1272724	3394,294442
6	17	54	432	0,1304631	3311,281121
7	20	54	432	0,1288406	3352,980349
8	22	54	432	0,1302404	3316,94313
9	25	54	432	0,1288409	3352,972542
10	27	54	432	0,1255588	3440,619057
11	30	54	432	0,1281887	3370,031836
12	32	54	432	0,1312668	3291,007322
13	35	54	432	0,1274115	3390,588762
14	37	54	432	0,1293236	3340,457581
15	40	54	432	0,1315209	3284,649056
16	42	54	432	0,1297556	3329,336075
17	45	54	432	0,1278491	3378,983505
18	47	54	432	0,1305085	3310,129225
19	50	54	432	0,1285653	3360,160168
20	52	54	432	0,1282007	3369,71639
21	55	54	432	0,1285727	3359,966774
22	57	54	432	0,1279083	3377,419605
23	60	54	432	0,1285265	3361,174544
24	62	54	432	0,1291136	3345,89075
25	63	54	432	0,1405451	3073,746434
	Rata- rata			0,12994321	3327,711283

B1.3 Pengujian 3 (Hari 1 Sore)

No.	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bit/s)
1	2	54	432	0,160524	2691,186365
2	7	54	432	0,1258727	3432,038877
3	10	54	432	0,1301637	3318,897665
4	12	54	432	0,1291659	3344,53598
5	15	54	432	0,1270686	3399,738409
6	17	54	432	0,1279519	3376,268738
7	20	54	432	0,1289733	3349,530484
8	22	54	432	0,1272056	3396,076902
9	25	54	432	0,1274247	3390,237529
10	27	54	432	0,1270118	3401,258781
11	30	54	432	0,1290877	3346,562066
12	32	54	432	0,1284657	3362,765314
13	35	54	432	0,1281407	3371,29421
14	37	54	432	0,1295482	3334,666171
15	40	54	432	0,128739	3355,6265
16	42	54	432	0,128765	3354,948938
17	45	54	432	0,1277223	3382,338088
18	47	54	432	0,1295323	3335,075499
19	50	54	432	0,1288245	3353,399392
20	52	54	432	0,1286397	3358,216787
21	55	54	432	0,1289841	3349,250024
22	57	54	432	0,1289249	3350,787939
23	60	54	432	0,1297002	3330,758164
24	62	54	432	0,1284227	3363,891275
25	63	54	432	0,0572754	7542,505159
	Rata- rata			0,12688538	3503,67421

B1.4 Pengujian 4 (Hari 2 Pagi)

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,115085	3753,74723
2	7	54	432	0,1278337	3379,390568
3	10	54	432	0,1293747	3339,138178
4	12	54	432	0,1286005	3359,240438
5	15	54	432	0,1271527	3397,48979
6	17	54	432	0,129588	3333,642004
7	20	54	432	0,1286329	3358,394314
8	22	54	432	0,1292865	3341,416157
9	25	54	432	0,1276368	3384,603813
10	27	54	432	0,1277164	3382,494339
11	30	54	432	0,1273485	3392,266104
12	32	54	432	0,1279304	3376,836155
13	35	54	432	0,1287267	3355,947135
14	37	54	432	0,1286911	3356,875495
15	40	54	432	0,1291888	3343,943128
16	42	54	432	0,12875	3355,339806
17	45	54	432	0,1278328	3379,41436
18	47	54	432	0,1285026	3361,799683
19	50	54	432	0,12844	3363,438181
20	52	54	432	0,1280021	3374,94463
21	55	54	432	0,1284639	3362,812432
22	57	54	432	0,1273737	3391,594968
23	60	54	432	0,1299033	3325,550621
24	62	54	432	0,1273017	3393,513205
25	63	54	432	0,1432885	3014,89652
	Rata- rata			0,1284261	3367,14917

B.1.5 Pengujian 5 Hari 2 Siang

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,078538	5500,52204
2	7	54	432	0,1288357	3353,107873
3	10	54	432	0,1292622	3342,04431
4	12	54	432	0,1286302	3358,464808
5	15	54	432	0,1291059	3346,090303
6	17	54	432	0,1287642	3354,969782
7	20	54	432	0,1290656	3347,1351
8	22	54	432	0,1285874	3359,582665
9	25	54	432	0,1274617	3389,253399
10	27	54	432	0,1280227	3374,401571
11	30	54	432	0,1289373	3350,465691
12	32	54	432	0,129065	3347,150661
13	35	54	432	0,1288334	3353,167734
14	37	54	432	0,1271501	3397,559263
15	40	54	432	0,1279252	3376,973419
16	42	54	432	0,1331309	3244,92661
17	45	54	432	0,1293656	3339,373064
18	47	54	432	0,1285442	3360,711724
19	50	54	432	0,1275422	3387,114226
20	52	54	432	0,1294478	3337,252545
21	55	54	432	0,1294292	3337,732135
22	57	54	432	0,1283121	3366,790817
23	60	54	432	0,1283344	3366,205787
24	62	54	432	0,1258515	3432,617013
25	63	120	960	0,0737544	13016,17259
	Rata- rata			0,12447588	3829,591405

B1.6 Pengujian 6 Hari 2 sore

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,040266	10728,65445
2	7	54	432	0,1254626	3443,257194
3	10	54	432	0,1275691	3386,399998
4	12	54	432	0,1273577	3392,021056
5	15	54	432	0,1280248	3374,34622
6	17	54	432	0,1260275	3427,823293
7	20	54	432	0,1267064	3409,456823
8	22	54	432	0,1268828	3404,716794
9	25	54	432	0,1277496	3381,615285
10	27	54	432	0,1271785	3396,80056
11	30	54	432	0,1269609	3402,622382
12	32	54	432	0,1280694	3373,171109
13	35	54	432	0,1277542	3381,493524
14	37	54	432	0,1263122	3420,097188
15	40	54	432	0,1282669	3367,977241
16	42	54	432	0,1268227	3406,330255
17	45	54	432	0,1268801	3404,789246
18	47	54	432	0,1266181	3411,834485
19	50	54	432	0,1277236	3382,303662
20	52	54	432	0,1267026	3409,559078
21	55	54	432	0,1277374	3381,938258
22	57	54	432	0,145908	2960,7698
23	60	54	432	0,1283344	3366,205787
24	62	54	432	0,1258515	3432,617013
25	63	54	432	0,0737544	5857,277668
	Rata- rata			0,1222769	3772,163135

B1.7 Pengujian 7 Hari 3 Pagi

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,039204	11019,28375
2	7	54	432	0,1253701	3445,797682
3	10	54	432	0,126599	3412,349229
4	12	54	432	0,1273234	3392,934842
5	15	54	432	0,1278608	3378,674308
6	17	54	432	0,1269158	3403,831517
7	20	54	432	0,128228	3368,998971
8	22	54	432	0,1256348	3438,53773
9	25	54	432	0,1273869	3391,243527
10	27	54	432	0,1255733	3440,221767
11	30	54	432	0,1269954	3401,698014
12	32	54	432	0,1271268	3398,181973
13	35	54	432	0,1282162	3369,309026
14	37	54	432	0,1277942	3380,435106
15	40	54	432	0,1292774	3341,651364
16	42	54	432	0,1264471	3416,448459
17	45	54	432	0,1302665	3316,278552
18	47	54	432	0,1262054	3422,991409
19	50	54	432	0,1279342	3376,735853
20	52	54	432	0,1302419	3316,904928
21	55	54	432	0,1278186	3379,789796
22	57	54	432	0,01638	26373,62637
23	60	54	432	0,1299033	3325,550621
24	62	54	432	0,1273017	3393,513205
25	63	54	432	0,1432885	3014,89652
	Rata- rata			0,12021173	4596,795381

B1.8 Pengujian 8 Hari 3 Siang

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,038366	11259,96976
2	7	54	432	0,1268449	3405,734089
3	10	54	432	0,1273309	3392,734992
4	12	54	432	0,1273746	3391,571004
5	15	54	432	0,1249685	3456,871132
6	17	54	432	0,155975	2769,674627
7	20	54	432	0,886383	487,3739681
8	22	54	432	0,1274525	3389,498048
9	25	54	432	0,1276182	3385,09711
10	27	54	432	0,1276036	3385,484422
11	30	54	432	0,1274008	3390,873527
12	32	54	432	0,1426019	3029,412652
13	35	54	432	0,079937	5404,255851
14	37	54	432	0,1276933	3383,10624
15	40	54	432	0,033841	12765,58021
16	42	54	432	0,0335374	12881,14549
17	45	54	432	0,079843	5410,618338
18	47	54	432	0,0858782	5030,380236
19	50	54	432	0,027499	15709,66217
20	52	54	432	0,0324977	13293,24845
21	55	54	432	0,0973833	4436,078876
22	57	54	432	0,1278384	3379,266324
23	60	54	432	0,1022343	4225,587694
24	62	54	432	0,068915	6268,591743
25	63	54	432	0,027894	15487,20155
	Rata- rata			0,1265965	6096,76074

B1.9 Pengujian 9 Hari 3 Sore

No	No. Sampel	Lenght (Byte)	bits	Delay (s)	Throughput (bits/s)
1	2	54	432	0,040446	10680,90788
2	7	54	432	0,1258646	3432,259746
3	10	54	432	0,12761	3385,314631
4	12	54	432	0,1269443	3403,06733
5	15	54	432	0,1257535	3435,292059
6	17	54	432	0,1262494	3421,79844
7	20	54	432	0,1267486	3408,32167
8	22	54	432	0,126764	3407,907608
9	25	54	432	0,1256832	3437,213566
10	27	54	432	0,125957	3429,741896
11	30	54	432	0,1271424	3397,765026
12	32	54	432	0,1271217	3398,318304
13	35	54	432	0,1273094	3393,307957
14	37	54	432	0,1265897	3412,599919
15	40	54	432	0,1269629	3402,568782
16	42	54	432	0,1264467	3416,459267
17	45	54	432	0,1277699	3381,078016
18	47	54	432	0,1276672	3383,797874
19	50	54	432	0,1281398	3371,317889
20	52	54	432	0,1257387	3435,696409
21	55	54	432	0,1280162	3374,572906
22	57	54	432	0,1260529	3427,132577
23	60	54	432	0,1270991	3398,922573
24	62	54	432	0,0152987	28237,6934
25	63	54	432	0,078648	5492,828807
	Rata- rata			0,11696096	4774,635381

B2. HASIL PENGUJIAN JARINGAN BERBASIS WEB

B2.1 Tabel Data Tes Website

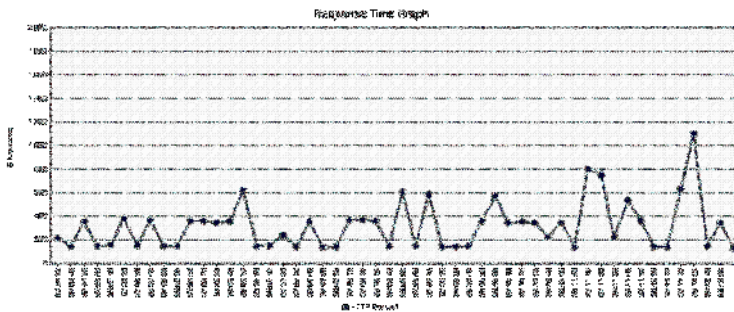
No.	Lenght (Byte)	Bits	Throughput (Bits/S)	Delay (S)
1	414	3312	45000	0,121
2	414	3312	57000	0,089
3	414	3312	61000	0,091
4	414	3312	7738	0,664
5	414	3312	64000	0,079
6	414	3312	64000	0,086
7	414	3312	64000	0,079
8	414	3312	1090	5,108
9	414	3312	33000	0,165
10	414	3312	641	8,009
11	414	3312	36000	0,15
12	414	3312	31000	0,177
13	414	3312	4397	1,266
14	414	3312	33000	0,152
15	414	3312	46000	0,112
16	414	3312	53000	0,104
17	414	3312	38000	0,134
18	414	3312	61000	0,09
19	414	3312	55000	0,1
20	414	3312	525	9,687
21	414	3312	6209	0,827
22	414	3312	55000	0,093
23	414	3312	32000	0,158
24	414	3312	2007	4
25	414	3312	29000	0,189
Rata-Rata			35184,28	1,25068

B3. HASIL PENGUJIAN KINERJA SERVER

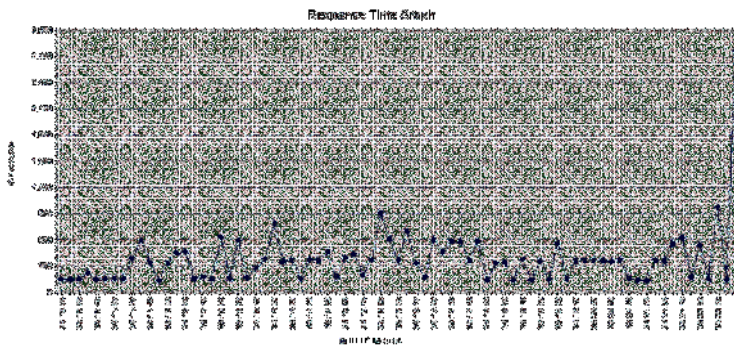
B3.1 Tabel Data Tes Server

Jumlah User	Label	Samples	Std. Dev	Error (%)	Throughput /sec
500	HTTP Request	5000	1853,92	0,78	9,65
	TOTAL	5000	1853,92	0,78	9,66
750	HTTP Request	7500	1971,6272	0,89	9,61
	TOTAL	7500	1971,6272	0,89	9,61
1000	HTTP Request	10000	1977,3105	0,89	9,81
	TOTAL	10000	1977,3105	0,89	9,81
1250	HTTP Request	12500	1914,491	0,832	9,87
	TOTAL	12500	1914,491	0,832	9,87
1500	HTTP Request	15000	1968,3807	0,89	9,90
	TOTAL	15000	1968,3807	0,89	9,90
2500	HTTP Request	25000	2.012	0,896	10
	TOTAL	25000	2.012	0,896	10
4000	HTTP Request	40000	2198	1,075	10
	TOTAL	40000	2198	1,075	10

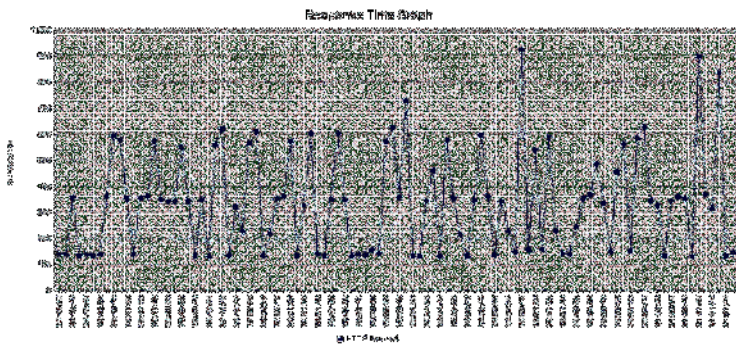
B3.2 Grafik Respon Time Kinerja Server User 500



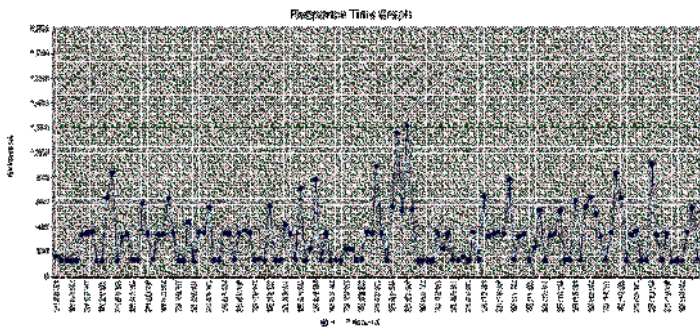
B3.3 Grafik Respon Time Kinerja Server User 750



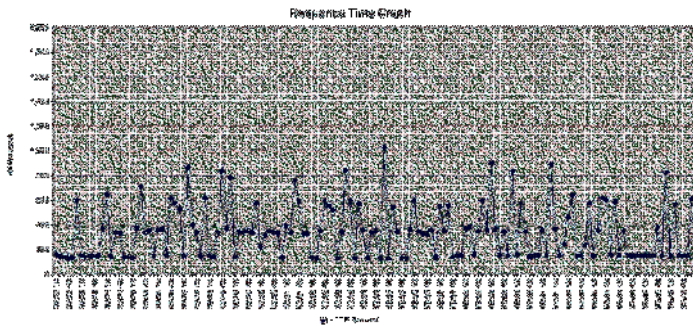
B3.4 Grafik Respon Time Kinerja Server User 1000



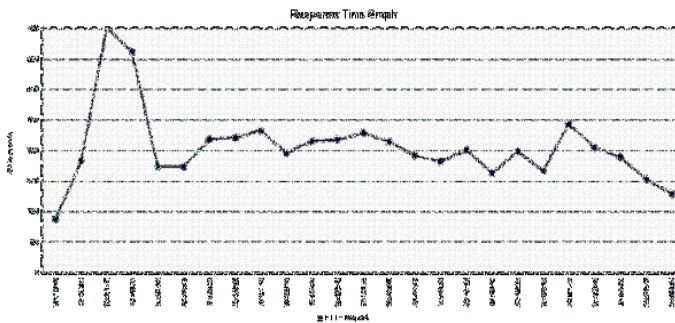
B3.5 Grafik Respon Time Kinerja Server User 1250



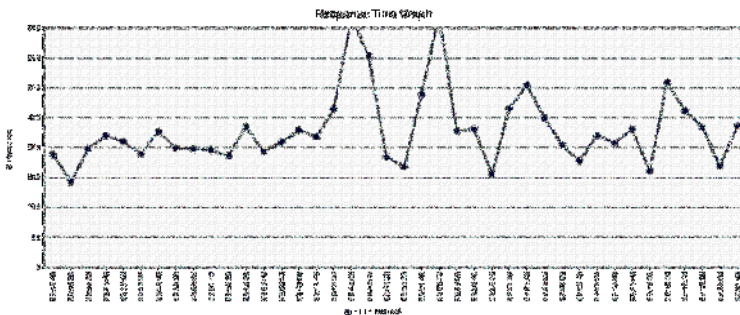
B3.6 Grafik Respon Time Kinerja Server User 1500



B3.7 Grafik Respon Time Kinerja Server User 2500



B3.8 Grafik Respon Time Kinerja Server User 4000



B3.9 Grafik CPU Load Server



LAMPIRAN C

LISTING PROGRAM

C.1 *Listing* program Borland Delphi 6 pada OBU

```
unit top;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, DB, ZAbstractRODataset, ZAbstractDataset, ZDataset,
  ZConnection,
  CPort, StdCtrls, Grids, DBGrids;

type
  TForm1 = class(TForm)
    Memo1: TMemo;
    ComboBox1: TComboBox;
    Button1: TButton;
    Label1: TLabel;
    ComPort1: TComPort;
    ZConnection1: TZConnection;
    ZQuery1: TZQuery;
    DBGrid1: TDBGrid;
    DataSource1: TDataSource;
    ZQuery2: TZQuery;
    Edit1: TEdit;
    Button2: TButton;
    Button3: TButton;
  procedure Button1Click(Sender: TObject);
  procedure ComPort1RxChar(Sender: TObject; Count: Integer);
  procedure Button2Click(Sender: TObject);
  procedure Button3Click(Sender: TObject);
```

```

private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.ComPort1RxChar(Sender: TObject; Count:
Integer);

var
    Str,s,s1,ID,sKat: String;
    SAldoAwal,xTarif:Double;

begin
    // CAption:='Data:'+IntToStr(Count);
    Memo1.Clear;
    if CCount>=16 then
        begin
            ComPort1.ReadStr(Str, Count);
            Memo1.Lines.Add(Str);
            ID:=Str;
            Edit1.Text:=Copy(Str,2,12);
            ZQuery1.SQL.Text:=
                'SELECT * FROM db_rfid WHERE id="'+Edit1.Text+'";
            ZQuery1.Close;
            ZQuery1.Open;

```

```

    if ZQuery1.RecordCount>0 then
    begin
    if ZQuery1.FieldByName('Kategori').Text='Pelajar' then
    begin
        sKat:='Pelajar';

    end
    else
    if ZQuery1.FieldByName('Kategori').Text='Dewasa' then
    begin
        sKat:='Dewasa';

    end
    else
    if ZQuery1.FieldByName('Kategori').Text='Anak-Anak' then
    begin
        sKat:='Anak-anak';

    end
    else
    if ZQuery1.FieldByName('Kategori').Text='Lansia' then
    begin
        sKat:='Lansia';

    end;

    if ZQuery1.RecordCount>0 then
    begin
    SaldoAwal:=ZQuery1.FieldByName('Saldo').AsInteger;
    Memo1.Lines.Add(
        '==>-Ini adalah kartunya
    '+ZQuery1.FieldByName('Nama').Text+
        #13#10' - Kategori '+sKat+
        #13#10' - Saldo Awal Rp. '+FormatFloat('0',SaldoAwal)+

```

```

        #13#10' - Silahkan melakukan Top-Up'+
        #13#10");
    end
else
Memo1.Lines.Add('Kartu belum terdaftar !');
end;
    end;
end;

```

```

procedure TForm1.Button1Click(Sender: TObject);

```

```

var

```

```

    Str,s,s1,ID,sCatatan,sKat: String;
    SAldoAwal,xTarif:Double;

```

```

begin

```

```

    if ZQuery1.RecordCount>0 then

```

```

    begin

```

```

        if ComboBox1.ItemIndex=0 then

```

```

        begin

```

```

            xTarif:=5000;

```

```

        end

```

```

        else if ComboBox1.ItemIndex=1 then

```

```

        begin

```

```

            xTarif:=10000;

```

```

        end

```

```

        else if ComboBox1.ItemIndex=2 then

```

```

        begin

```

```

            xTarif:=20000;

```

```

        end

```

```

        else if ComboBox1.ItemIndex=3 then

```

```

        begin

```

```

            xTarif:=25000;

```

```

end
    else if ComboBox1.ItemIndex=4 then
begin
xTarif:=50000;
end
    else if ComboBox1.ItemIndex=5 then
begin
xTarif:=100000;
end;
SaldoAwal:=ZQuery1.FieldByName('Saldo').AsInteger;
Memo1.Lines.Add(
'- Top-up Rp. '+FormatFloat('0',xTarif)+
#13#10' - Saldo Akhir Rp.
'+FormatFloat('0',SaldoAwal+xTarif)+
#13#10");

Zquery2.SQL.Text:=
'UPDATE db_rfid SET Saldo=Saldo+'
+FLoatToStr(xTarif)+
' WHERE ID="'+Edit1.Text+""';
ZQuery2.ExecSQL;

Memo1.Lines.Add('====>Top-Up berhasil !');

sCatatan:=FLoatToStr(xTarif);
Zquery2.SQL.Text:=
'INSERT INTO tb_logtopup(CardID,Topup)
VALUES(""+Edit1.Text+"" ,""+sCatatan+"");
ZQuery2.ExecSQL;

end;
end;

```

RIWAYAT HIDUP



Ulfi Romadhani Rosadiana, lahir di Surabaya 21 Januari 1996. Menamatkan pendidikan dasar di SDN Pesu 1 Maospati Magetan pada tahun 2008 kemudian meneruskan sekolah menengah pertama di SMPN 1 Barat Magetan dan lulus tahun 2011. Sekolah menengah atas dilanjutkan di SMAN 3 Madiun dan lulus pada tahun 2014. Pendidikan dilanjutkan pada tingkat strata satu (S1) di jurusan Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2014 hingga 2018 dengan program studi

Telekomunikasi Multimedia. Penulis dapat dihubungi melalui alamat email : ulfiromadhani.21@gmail.com